PC-9801 3次元CG作成システム MODE

太田 昌孝 監修
アスキー書籍編集部 編

本格的CGが
あなたのPC-9801で楽しめる！

アスキー出版局

Art Gallery

矢口　英夫

横山　弥生

飯田　恒夫

ここにある作品はMODEで作成し、フレームバッファとフレームバッファ専用ソフトを使って表示しています。

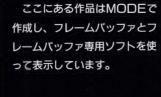
中山　洋三

鈴木　敏晃

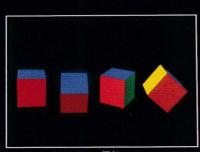
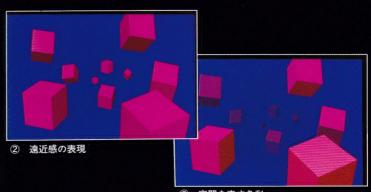
森本　忠則

渡辺　和年

五島　理江

使用機材：MODEシステム＋SuperFrame2＋SuperTableau Ver.2.0(またはRAY TREK2)

CGデザインのさまざまな表現

① X、Y、Z軸上での回転

② 遠近感の表現

③ 空間を表す色彩

④ 平行光源を使った光の明暗

⑤ 点光源を使った光の明暗

⑥ 線の構成(作：東海　徳亮)

⑦ 構成——軽い

⑧ 構成——重い

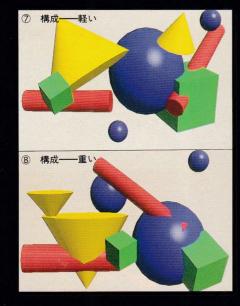
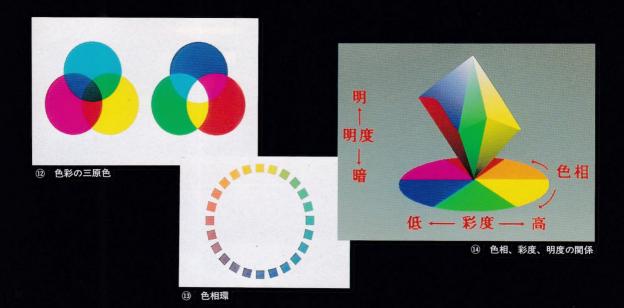
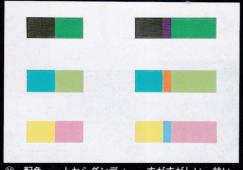
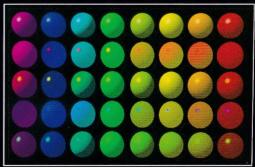
⑨ 泣いた顔(作：兼子　美弥子)

⑩ 笑った顔

⑪ 怒った顔

⑫ 色彩の三原色

⑬ 色相環

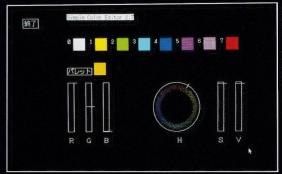
⑭ 色相、彩度、明度の関係

⑮ 配色——上からダンディー、すがすがしい、甘い

⑯ 光源色による変化——上から白、赤、黄、青、緑の光を当てた

⑰ 質感の変化

⑱ 陶器風と金属風の壺

アニメーション―付属ファイルの実行例―

⑲　サンプル
（SAMPLE.MOD、73ページ参照）

⑳　サンプル
（5 コマ目、73ページ参照）

㉑　カラーエディタ（CEDIT.EXE、157ページ参照）

㉓ ㉔ ㉕ ㉖ ㉗

ロボット（ROBOT2.MOD、168ページ参照）

㉘ ㉙ ㉚ ㉛

㉒　木（TREE3.MOD、163ページ参照）

㉜　グラス（GLASS3.MOD）

ここにあるのは、すべてフレームバッファを使わない標準構成で表示しています。

PC-9801
3次元CG作成システム
MODE

太田 昌孝 監修
アスキー書籍編集部 編

アスキー出版局

商　標

MS、MS-DOS は米国 Microsoft 社の登録商標です。その他、本書で登場するシステム名、製品名は一般に各開発メーカーの登録商標です。なお、本文中では TM、Ⓡマークは明記していません。

☆本パッケージに収められているデータのうち YUKI.EXE は、LHA の自己解凍ファイルで提供しています。

カバーデザイン　：　郷 啓子
カバーCG 作成　：　横山 弥生、渡辺 和年、五島 理江、中山 洋三、
矢口 英男（青山 CG スクール・メロン）

はじめに

CG(コンピュータグラフィックス)は、単なるお絵描きツールではありません。分子モデルや太陽系など手書きでは正確に描きにくいものや、街/ビル/部屋のレイアウトなどのように描き直しが大変なもの、また色や模様をいろいろと差し替えるデザイン(テクスチャー)や光の当たり方を変更する絵、さらにゴルフボールの弾道や構造物の中を自由に見回すようなシミュレーション(アニメーション)などの表現もCGならではです。

CGは『スターウォーズ』、『トロン』、『ターミネーター2』などの映画制作の現場やCFではすでに欠かせないものになっていますし、アニメーション、シミュレーション、ファインアート、デザイン、画像処理、CAIとさまざまな分野で使われています。

またCGにはこういった現場で活躍するプロ用のツールとしてだけでなく、ホビーとしての楽しみ方もあります。しかし、お絵描きツールを除いて安価なアプリケーションが少なく、また自作するには少々複雑なプログラムです。

本パッケージで紹介する〝MODE〟システムとは『応用グラフィックス』(弊社刊)で紹介した汎用的なCGツール〝MO｜RE〟システムを発展させた3次元CG作成システムです。

MO｜REとは、MOdellingとMOtionとMOtion-designの結果をMS-DOSのパイプ機能でREnderingとREcordingサブシステムに渡してCGを作成するという構成からきたもので、これはMO｜RE>GRAPHICS(もっとグラフィックスを!)という願いから名付けられました。MODEとは、MO｜re Display Editor(MO｜REのディスプレイエディタ版)という意味です。

MODEでは、ポリゴンを作る〝ポリゴンエディタ〟機能、作ったポリゴンを配置する〝ツリーエディタ〟機能、動きを作る〝モーションエディタ〟機能などを使って編集し、カラーアニメーション表示ができます。また、ファイル形式は〝MO｜RE〟と互換なので、〝MO｜RE〟用のツールやファイルはそのままお使いになれます。

〝MODE〞システムは、特別なハードウェアを用意しなくてもお手持ちのパソコン(PC-9801)で手軽に本格的CGが楽しめます。ただし、お絵描きツールとは違い、CGの基礎的な知識が必要ですから注意してください。

本パッケージは5インチ2HD/3.5インチ2HDの両メディア付きで、便利なコマンドと豊富なサンプルデータ・ファイルを多数収録しました。また読者が自分でプログラミングする際に参考となるようにC言語のソースリストも収録しました。

『PC-9801　3次元CG作成システムMODE』を制作するに当たっては、『応用グラフィックス』の著者であり、MO｜REシステムの開発者でもある東京工業大学の太田昌孝氏に監修をお願いしました。また、MODEプログラムの開発は、出口哲生氏と丸川一志氏、各種ツールの開発は、石井秀治氏、田口景介氏、河村博之氏が担当しました。さらに、当初よりMO｜REシステムを導入し、作品制作やMO｜REの拡張を行っている青山CGスクール・メロンの横山弥生氏に「1章　数値と感性によるCGデザイン」の執筆と作品提供を、同じく島田信吾氏に「2章　アニメーション」と「Appendix F　フレームバッファ」の執筆をしていただきました。ここに感謝の意を表します。

アスキー書籍編集部

本書を読む前に

■必要な機器構成

本パッケージを使用するには最低限、以下のハードウェア、ソフトウェアが必要です。

●パソコン

日本電気 PC-9800 シリーズ(PC-9801/E/F/M/U/UV/VM/DO/DO+、PC-98XA/LT/HA を除く)で使用できます。98NOTE は、カラーディスプレイが接続できる場合に限って使用可能です。

PC-98XL/XL2/RL などのハイレゾモードとノーマルモードが切り替えられる機種では、ノーマルモードでのみ利用可能です。

メモリのフリーエリアが 380K バイト以上必要です。

●カラーディスプレイ

●マウス

●MS-DOS

本パッケージには「MS-DOS」が含まれていませんので、日本電気製の MS-DOS Ver.2.0、Ver.3.1、Ver.3.3x のいずれかが必要です。

■必要な予備知識

本文中ではCGの技術的な解説はしていません。本パッケージをお使いになるには、事前にCGの入門書を1冊読んだ程度の基礎的な知識が最低限必要です。

初心者の方は、たとえば以下のような本(弊社刊)を読んでから始められることをお勧めします。

●入門グラフィックス

描画アルゴリズム、色と光、移動/回転/スケーリング変換、透視/投影変換、ワイヤフレームモデル、クリッピングなど、グラフィックスの基礎知識から応用テクニックまでをひと通り解説。

●実習グラフィックス

モデリング変換、視野変換、陰線処理、曲線/曲面近似、CADについて解説。特に陰線処理のアルゴリズムと手法に関して分かりやすく紹介。

●応用グラフィックス

サーフェイス/ソリッドモデル、自由曲面、フラクタル、隠面処理、シェーディング、アンチエイリアシング、モーションデザインなどについて解説。さらに、リアルな3次元グラフィックスを展開するためのC言語プログラミングなども紹介。

●最新　3次元コンピュータグラフィックス

最新のCG原理や手法を、むずかしいプログラムや数式はいっさい使わず、豊富なカラー作品や図版によって分かりやすく解説。

"MODE" システムの構成

"MODE"システムは、ポリゴンを作る"ポリゴンエディタ"機能、作ったポリゴンを配置する"ツリーエディタ"機能、動きを作る"モーションエディタ"機能などを持つメインプログラムとカラーエディタや平面エディタなどのツールで構成されています。

■メインプログラム

メインプログラムは、以下のファイルで構成されています。MODE.EXE 以外は単体でも使用できます。

DRAW.EXE	イメージファイルを生成
DITHER.EXE	RENDER が出力したイメージファイルを画面に表示
GSAVE.EXE	画面上のイメージファイルのセーブ
GLOAD.EXE	画面上のイメージファイルのロード
MODE.EXE	モデルファイルとモーションファイルの作成
MODE.TXT	ツリーエディタで使えるツールの説明
MODEL.EXE	モデルファイルから RENDER 用のデータを生成
RENDER.EXE	MODEL.EXE が出力したデータからイメージファイルを生成

■ツール

ツールは以下のファイルで構成されています。

CEDIT.EXE	色情報の生成
CONE.EXE	円錐のポリゴンデータを生成
CUBE.EXE	立方体の角をカットしたポリゴンデータを生成
SPHERE.EXE	球のポリゴンデータを生成
POLHD.EXE	正多面体のポリゴンデータを生成
TUBE.EXE	円柱のポリゴンデータを生成
ROT.EXE	ポリゴンデータをもとに回転体を作成
THICK.EXE	ポリゴンデータに厚みを付加
CLS3.EXE	グラフィック画面の消去
MOTION.EXE	モーションファイルからパラメータファイルを作成
PLANE.EXE	選択点の順番でポリゴンを生成

本書の構成

インストールに続いてPart 1ではCGの基礎、Part 2ではMODEシステムの基本的な使い方、Part 3ではシェーディングについて解説しています。またAPPENDIXではテキストエディタを使った応用例、ソースファイルのコンパイル方法とコマンドリファレンスを掲載しています。

なお、本書中に解説のあるコマンドとサンプルデータは、すべて付属のディスクパッケージに収録しています。付属のディスクの構成は、「インストール」をご覧ください。

■インストール

ディスクからのインストール方法を3つの場合に分けて説明します。添付のインストールプログラムを使わずにインストールをされる方は、このうち「用意するもの」と「MODEの環境設定」をお読みください。

■Part 1　CGの基礎

CGを始めるにあたって、絵画との違いやCGデザインの心得、物体の組み立て方について解説します。すでに基礎的な知識のある方は、Part 2から始められても結構です。

1章　数値と感性によるCGデザイン

CGを使ってデザインをするには、空間の把握や部品の配置、配色、光源とその方向などの知識が必要です。1章ではCGデザインに必要な考え方について解説します。

2章　アニメーション

3次元CGアニメーションをデザインするには、まず物体の動きを分析しなければなりません。2章ではコンピュータを使ったアニメーションデザインの基礎について解説します。

■Part 2　MODEシステム

MODEシステムの基本的な使い方について説明します。まず静止画を作るためにポリゴンエディタとツリーエディタを説明し、次に動画を作るためのモーションエディタを説明しています。そして最後に2つの作品を作っていきながら、その制作過程を解説します。

3章　MODEの概要

添付のディスクに収録されているサンプルデータを使って、MODEの起動と終了、MODEメニューの説明、3つのエディタの関係、図形の表示について説明します。

4章　ポリゴンエディタ

MODEの機能の1つに物体を作ったり動きの確認ができる〝ポリゴンエディタ〟があります。ここではポリゴンエディタを使った物体の作り方について説明します。

5章　ツリーエディタ

ポリゴンエディタで作った物体を配置したり色指定したりする機能が〝ツリーエディタ〟です。ここではツリーエディタを使った物体の配置やツールの使い方について説明します。

6章　モーションエディタ

動きをデザインする機能が〝モーションエディタ〟です。ここではモーションエディタを使って動かす部分や量、コマ数の指定方法について説明します。

7章　ファイルの切り替え

複数のファイルを開いているとき、すばやく特定のファイルに切り替える機能が〝ファイルの切り替え〟です。ここではその使用方法について説明します。

8章　モデリング実践

木とロボットを取り上げ、実際に作品を作ってみます。

■Part 3　シェーディング

物体や背景に色を付けるにはいくつかの約束ごとがあります。9章では色指定の方法について説明し、10章では8章で作った2つの作品に色を付けます。

9章　色情報

物体や背景に色を付けるにはツリーエディタ上でいくつかの指定をしなければなりません。ここでは色指定に必要な視野モデル用コマンド、背景色の指定コマンド、光源モデル用コマンド、表面モデル用コマンド、シェーディングパラメータ指定用コマンドについて説明します。また、簡単に背景色を作成できるカラーエディタについても解説します。

10章　シェーディング実践

8章で作った木とロボットに、それぞれ色を付けます。

■Appendix

他のエディタを使った応用例や、パッケージにある、すべての実行ファイルのコマンドリファレンスを掲載しています。

Appendix A　テキストエディタを使ったデータ編集

MODE のモデリングデータはテキストファイル形式で保存されるので、ツリーエディタの代わりにテキストエディタでも編集できます。ここではテキストエディタを使ったデータの編集について説明します。

Appendix B　ソースファイルのコンパイル

本パッケージに収録したプログラムには、すべてソースファイルが付属しています。ここでは添付のソースファイルのコンパイル方法について説明します。

Appendix C　クイックリファレンス ―― MODE システム

〝ポリゴンエディタ〞、〝ツリーエディタ〞、〝モーションエディタ〞、〝ファイルの切り替え〞の各機能で使う MODE システムのコマンドをまとめてあります。

Appendix D　コマンドリファレンス ―― ツリーエディタで使えるツール

PLANE コマンドを始めとしたポリゴン作成を支援するツール 9 本の使い方を説明しています。

Appendix E　コマンドリファレンス ―― コマンドラインから使うツール

MODE システムの機能の一部を単体のコマンドとして使う場合について説明しています。これらは MS-DOS のコマンドラインから使います。

Appendix F　フレームバッファ

滑らかなシェーディングをするためにはフレームバッファが有効です。ここではフレームバッファについて解説します。

目　次

Part2 MODEシステム

Part3 シェーディング

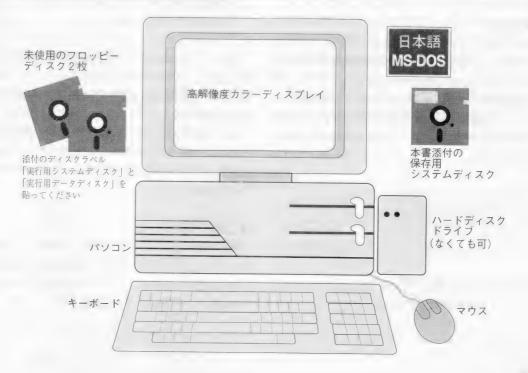
APPENDIX

インストール

本書に付属のフロッピーディスク(保存用ディスクと呼びます)には、本ソフトウェアを動かすために必要な「基本ソフトウェア」である「MS-DOS」は含まれていません。本ソフトウェアを実行するためには、MS-DOSが別途必要になります。

以下では、このMS-DOSを組み込んだMODEの実行用ディスクの作成方法を説明します。

作成の前に必要なハードウェアとソフトウェアを確認してください。

用意するもの

●パソコン

本書に付属のフロッピーディスクに入っているソフトウェアは、次のパソコンで使用できます。日本電気 PC-9800 シリーズのカラー表示可能な機種(PC-9801/E/F/M/U/UV/VM/DO/DO+、PC-98XA/LT/HA を除く)。98NOTE は、カラーディスプレイが接続できる場合に限って使用可能です。

PC-98XL/XL2/RL などのハイレゾモードとノーマルモードが切り替えられる機種では、ノーマルモードでのみ利用可能です。

なお、ソフトウェアの実行にはメモリのフリーエリアが 380K バイト以上必要です。

●ディスプレイ

本ソフトウェアを使用するには、640×400 ドット表示可能な高解像度カラーディスプレイ(アナログ RGB でもデジタル RGB でも可)が必要です。

●マウス

本ソフトウェアを利用するには、マウス(バスマウスでもシリアルマウスでも可)が必要です。マウスがないと利用できない機能があります。

実行用ディスクの作成においてはマウスインターフェイスの種別を選択する部分がありますが、内蔵されているマウスインターフェイスの形を見て判断してください。

シリアルマウス　　バスマウス

※ MS-DOS に付属の MOUSE.SYS などのマウスドライバは必要ありません。

●キーボード

●ハードディスクドライブ

ハードディスクドライブに本ソフトウェアを組み込んで利用する場合に必要です。

アニメーションを実行する場合は、設定したコマ数によっては1枚のディスクに入り切らないことがあります。できればハードディスクでのご使用を推奨します。

必要な空き容量は、「ハードディスクに実行用ディスクを作成」を参照してください。

● MS-DOS

本ソフトウェアが対応している MS-DOS のバージョンは、次のうちのいずれかです。

・日本電気製の MS-DOS Ver.2.0
・日本電気製の MS-DOS Ver.3.1
・日本電気製の MS-DOS Ver.3.3x

なお、フロッピーディスクで実行用ディスクを作るときには、MS-DOS の FORMAT コマンド(FORMAT.EXE か FORMAT.COM の入っているディスク)が必要です。

また、フロッピーディスクのバックアップを作るときには、MS-DOS の DISKCOPY コマンド(DISKCOPY.EXE か DISKCOPY.COM の入っているディスク)が必要です。

●コンパイラ

本パッケージにはソースリストが付属しています。ソースリストをコンパイルするには米国 Microsoft 社の Microsoft C Professional Development System Version 6.0 が必要です。

●アセンブラ
　添付のディスクに含まれるアセンブラのソースリストをアセンブルするためにはMicrosoft　Macro Assembler Version 5.10が必要です。

● RAMディスク
　RAMディスクがあれば、より快適に本ソフトウェアがご使用になれます。

●数値演算処理プロセッサ
　数値演算処理プロセッサがあれば、本ソフトウェアの描画速度を向上できます。ご利用になるときは、ご使用のパソコンに合ったものをご用意ください。

●フレームバッファ
　フレームバッファがあれば、表示色を拡張して滑らかなシェーディングができます。ご利用になるときは、ご使用のパソコンに合ったものをご用意ください。
　フレームバッファについては「Appendix F　フレームバッファ」でも紹介しています。

●未使用のフロッピーディスク2枚
　ご自分のパソコンで使えるサイズ(5インチか3.5インチのどちらか)で2HDのものを用意してください。「実行用システムディスク」、「実行用データディスク」を作成するために使います。この2枚のディスクに貼るための予備のディスクラベルが、本書添付の5インチディスクのエンベロープに入っています。まず、このラベルを用意したディスクに貼ってください。

※ハードディスクドライブに実行用ディスクを作る場合には、フロッピーディスクを用意する必要はありません。

●本パッケージ添付の「保存用ディスク」
　本パッケージには外カバー(ビニール装)の内側の表紙側に3.5インチ2HD、背表紙側に5インチ2HDのフロッピーディスクが各2枚付いています。コマンドやデータを収録した「保存用システムディスク」とソースファイルを収録した「保存用データディスク」があり、これら2枚ずつのディスクの内容は同じです。以下の「実行用ディスクの作成」には、このうち「保存用システムディスク」だけを使用します。ご使用のパソコンに合ったサイズのディスクを選んでお使いください。
　「保存用ディスク」はバックアップをとっておきましょう(「ディスクのバックアップ」参照)。

実行用ディスクの作成

「実行用ディスク」として、「実行用システムディスク」、「実行用データディスク」の２枚を作ります。

さて、ここでは次の３通りの場合の説明をします。

①フロッピーディスクのみのシステム構成で実行用ディスクを作成する場合
②ハードディスクが含まれたシステム構成で実行用ディスクを作成する場合
③ハードディスクが含まれたシステム構成で、ハードディスクからMS-DOSを立ち上げ、フロッピーディスクの実行用ディスクを作成する場合

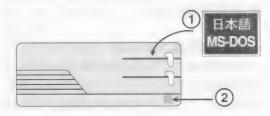
ここからは、自分が作りたい実行用ディスク(①〜③)に合った項目で、作業を進めてください。

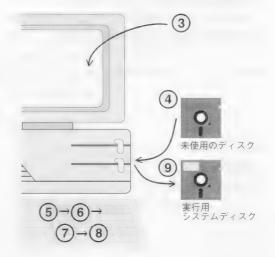
※MS-DOSを起動し、「保存用システムディスク」を入れてinst⏎と入力すると、次のメニューが表示されます。
※プログラムを起動するとディスクに書き込みを行いますので、実行用ディスクにはライトプロテクトシールを貼らないでください。

```
            《実行用ディスク作成メニュー》
------------------------------------------------------------
実行用ディスクを作成します。次のいずれかを選択してください。
<バスマウスを使用>
        1：「ﾌﾛｯﾋﾟｰﾃﾞｨｽｸ」のみのシステム構成で実行用ディスクを作成
        2：「ﾊｰﾄﾞﾃﾞｨｽｸ」に実行用ディスクを作成
        3：「ﾊｰﾄﾞﾃﾞｨｽｸ」から立ち上げ、「ﾌﾛｯﾋﾟｰﾃﾞｨｽｸ」に実行用ディスクを作成
<シリアルマウスを使用>
        4：「ﾌﾛｯﾋﾟｰﾃﾞｨｽｸ」のみのシステム構成で実行用ディスクを作成
        5：「ﾊｰﾄﾞﾃﾞｨｽｸ」に実行用ディスクを作成
        6：「ﾊｰﾄﾞﾃﾞｨｽｸ」から立ち上げ、「ﾌﾛｯﾋﾟｰﾃﾞｨｽｸ」に実行用ディスクを作成
------------------------------------------------------------

※「ﾌﾛｯﾋﾟｰﾃﾞｨｽｸ」に実行用ディスクを作成するときは、あらかじめシステム用
とデータ用に初期化されたディスクをご用意ください。

A>
```

画面１ インストールメニュー画面(INST.BAT)

▶▶フロッピーディスクのみのシステム構成で実行用ディスクを作成

●システム用ディスクの初期化

①MS-DOSシステムディスク(FORMAT.EXEかFORMAT.COMのFORMATコマンドが入っているもの)をドライブAに入れる。

※本体内蔵ディスクドライブのみを使用の場合は、1と書かれているドライブがドライブA、2と書かれているドライブがドライブBになります。
※画面に出るメッセージはMS-DOSのバージョンによって多少異なります。

② POWER(電源)スイッチを入れる。
③ MS-DOS が立ち上がったことを確認。
④ドライブ B に未使用のディスクを入れる。
⑤ MS-DOS のコマンドラインで次のように入力。

```
A>format b: /s
```

画面2　システム用フォーマット

以下の一連の初期化作業が始まります。

```
A>format b: /s
Format Version 4.60

新しいディスクをドライブ B: に挿入し
どれかキーを押してください
```

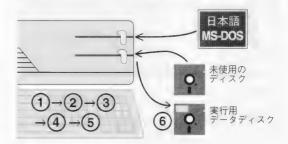
⑥ ↵ を押す。

```
A>format b: /s
Format Version 4.60

新しいディスクをドライブ B: に挿入し
どれかキーを押してください

ディスクのタイプは　1 : 2DD(640KB)  2 : 2HD(1MB)  =
```

⑦ 2 ↵ を入力。

```
A>format b: /s
Format Version 4.60

新しいディスクをドライブ B: に挿入し
どれかキーを押してください

ディスクのタイプは　1 : 2DD(640KB)  2 : 2HD(1MB)  = 2

目的のディスクは 1MB FD です

フォーマットが終了しました
0  10  20  30  40  50  60  70  80  90  100
                                            (%)
システムを転送しました

  1250304 バイト　全ディスク容量
   122880 バイト　システム領域
  1127424 バイト　使用可能ディスク容量

別のディスクをフォーマットしますか(Y/N)
   C1  CU  CA  S1  SU    VOID  NWL  INS  REP  ^Z
```

⑧ N ↵ を入力。
⑨ドライブ B のディスクを取り出し、「実行用システムディスク」のラベルを貼る。

●データ用ディスクの初期化

①「システム用ディスクの初期化」の①〜④の作業を行う。
② MS-DOS のコマンドラインで次のように入力。

```
A>format b:
```

画面3　データ用フォーマット

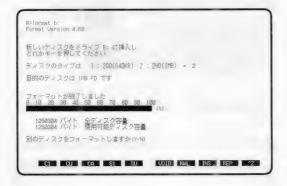
以下の一連の初期化作業が始まります。

```
A>format b:
Format Version 4.60

新しいディスクをドライブ B: に挿入し
どれかキーを押してください
```

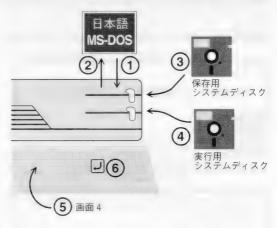
③[↵]を押す。

```
A>format b:
Format Version 4.60

新しいディスクをドライブ B: に挿入し
どれかキーを押してください

ディスクのタイプは   1 : 2DD(640KB)  2 : 2HD(1MB)  = ▮
```

④2[↵]を入力。

```
A>format b:
Format Version 4.60

新しいディスクをドライブ B: に挿入し
どれかキーを押してください

ディスクのタイプは   1 : 2DD(640KB)  2 : 2HD(1MB)  = 2

目的のディスクは 1MB FD です

フォーマットが終了しました
0  10  20  30  40  50  60  70  80  90  100
                                            (%)

   1250304 バイト  全ディスク容量
   1250304 バイト  使用可能ディスク容量

別のディスクをフォーマットしますか(Y/N)

 C1  CU  CA  S1  SU   VOID  NWL  INS  REP  ^Z
```

⑤N[↵]を入力。

⑥ドライブ B のディスクを取り出し、「実行用データディスク」のラベルを貼る。

●実行用システムディスク作成手順

① MS-DOS を立ち上げる。

②ドライブ A から MS-DOS のシステムディスクを取り出す。

③ドライブ A に本書添付の「保存用システムディスク」を入れる。

④ドライブ B に「実行用システムディスク」のラベルを貼ったディスクを入れる。

⑤ MS-DOS のコマンドラインで次のように入力。

以下バスマウスの場合で説明します。シリアルマウスの場合は、最初のキーボード入力が違う(4[↵]と入力)だけです。

```
A>1
```

画面 4　キーボード入力(1.BAT)

これで「実行用ディスク作成プログラム」が起動されますので、画面の指示にしたがって以下の手順どおりにインストールを進めてください。

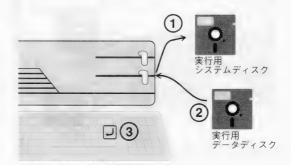
```
実行用システムディスクを作成します。

「実行用システムディスク」のラベルを貼ってあるディスクをBドライブに
入れてください。

もし都合が悪ければここで CTRL-C （コントロールキーを押しながら "c"
または "C" ）を押して処理を中止してください。
----------------------------------------------------------------------
準備ができたらどれかキーを押してください.
```

画面 5 1.BAT

⑥ [↵] キーを押す。作業を中止したい場合は [STOP] キー（または [CTRL] ＋ [C] ）を押す。

```
実行用システムディスクの作成を終わりました。
Bドライブのディスクを取り出してください。
次に実行用データディスクを作成します。
Bドライブに「実行用データディスク」のラベルを貼ってあるディスクを
入れてください。
----------------------------------------------------------------------
準備ができたらどれかキーを押してください.
```

これで「実行用システムディスク」の作成が終わりました。続いて「実行用データディスク」を作成します。

●実行用データディスク作成手順

①ドライブＢのディスクを抜く。

②ドライブＢに「実行用データディスク」のラベルを貼ってあるディスクを入れる。

③ [↵] キーを押す。

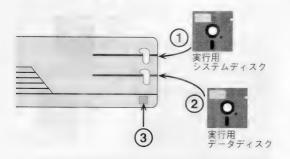
```
実行用システムディスクの作成を終わりました。
Bドライブのディスクを取り出してください。
次に実行用データディスクを作成します。
Bドライブに「実行用データディスク」のラベルを貼ってあるディスクを
入れてください。
----------------------------------------------------------------------
準備ができたらどれかキーを押してください.
A:¥SAMPLE¥PYRAMID.MOD
A:¥SAMPLE¥SAMPLE.000
A:¥SAMPLE¥SAMPLE.010
A:¥SAMPLE¥SAMPLE.MOD
A:¥SAMPLE¥SAMPLE.MOT
        5 個のファイルをコピーしました.
```

各種ファイルをコピーして次の画面を表示します。

```
実行用データディスクの作成が終わりました。
Aドライブの「保存用システムディスク」を取り出してください。
[RESET]ボタンを押して立ち上げ直してください。
----------------------------------------------------------------------
A>
```

これで「実行用データディスク」の作成が終わりました。

④「保存用システムディスク」をドライブＡから取り出す。

●プログラムの起動

①ドライブAに「実行用システムディスク」を入れる。
②ドライブBに「実行用データディスク」を入れる。
③POWER(電源)を入れる(電源が入っている場合はRESETボタンを押す)。

※MS-DOSのコマンドラインでmode⏎と入力すればMODEが起動しますが、この操作は3章をご覧になってから行ってください。

▶▶ハードディスクに実行用ディスクを作成

この作業を始める前に、次の事項をかならず確認してください。

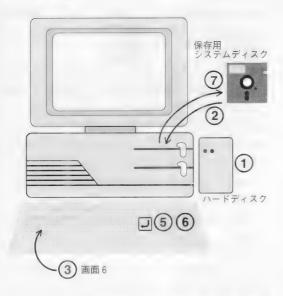
●ハードディスクの空き容量

本ソフトウェアをハードディスクに組み込むには、以下の表に示した「空き容量」が必要です。空き容量が不足している場合には、組み込みが正常に行われませんので、下表で指定されている容量以上の空きを確保したうえで作業にかかってください。

ハードディスク容量	インターフェイス規格	必要な空き容量
20Mバイト	SASI(8Kクラスタ)	約1100Kバイト
40Mバイト	SASI(16Kクラスタ)	約1500Kバイト
64Mバイト以下	SCSI(2Kクラスタ)	約800Kバイト
64Mバイト以上	SCSI(4Kクラスタ)	約900Kバイト

●ディレクトリ

この「実行用ディスク作成プログラム」は、ハードディスクドライブに"¥MODE"というディレクトリを作成(ディレクトリ名は固定)して本ソフトウェアに必要なプログラムをコピーし、"¥MODE¥DATA"というディレクトリを作成して本ソフトウェアに必要なデータファイルをコピーするようになっています。

●作業手順

①ハードディスクドライブからMS-DOSを立ち上げる。

②本書添付の「保存用システムディスク」を、いずれかのフロッピーディスクドライブに入れる。

ここでは、ハードディスクドライブ(ドライブA)からMS-DOSを立ち上げてドライブBに「保存用システムディスク」を入れて作業を進める例で説明します。

③キーボードから次のように入力。

以下バスマウスの場合で説明します。シリアルマウスの場合は、最初のキーボード入力が違う(5 [↵] と入力)だけです。

```
A>2 a:
```

画面6 キーボード入力(2.bat)

※ハードディスクをドライブAとする場合、「A：」は省略できます。

④こうすると、「実行用ディスク作成プログラム」が起動され、次の画面が表示されます。

```
ハードディスクに実行用ディスクを作成します。

ハードディスクをa:ドライブとして作業を始めます。

a:¥modeというディレクトリを作り、プログラムをコピーします。
次にa:¥mode¥dataというディレクトリを作り、データをコピーします。

フロッピーディスクドライブに「保存用システムディスク」を入れてください。

もし都合が悪ければここで CTRL-C（コントロールキーを押しながら“c”
または“C”）を押して処理を中止してください.
------------------------------------------------------------
準備ができたらどれかキーを押してください.
```

画面7 2.BAT

⑤ [↵] を押す。作業を中止したい場合は [STOP] キー(または [CTRL] + [C])を押す。

```
ハードディスクに実行用ディスクを作成します。

ハードディスクをa:ドライブとして作業を始めます。

a:¥modeというディレクトリを作り、プログラムをコピーします。
次にa:¥mode¥dataというディレクトリを作り、データをコピーします。

フロッピーディスクドライブに「保存用システムディスク」を入れてください。

もし都合が悪ければここで CTRL-C（コントロールキーを押しながら“c”
または“C”）を押して処理を中止してください.
------------------------------------------------------------
準備ができたらどれかキーを押してください.
C:¥BIN¥CEDIT.EXE
C:¥BIN¥CLS3.EXE
C:¥BIN¥CONE.EXE
C:¥BIN¥CUBE.EXE
C:¥BIN¥DITHER.EXE
C:¥BIN¥DRAW.EXE
C:¥BIN¥GLOAD.EXE
C:¥BIN¥GSAVE.EXE
```

各種ファイルをコピーして次の画面を表示します。

```
config.sysとautoexec.batをコピーします。
もとの内容は*.orgとして保存します。

もし都合が悪ければここで CTRL-C （コントロールキーを押しながら “c”
または “C” ）を押して処理を中止してください.
------------------------------------------------------------------------
準備ができたらどれかキーを押してください.
```

⑥ [⏎] を押す。作業を中止したい場合は [STOP] キー（または [CTRL] ＋ [C] ）を押す。

⑦「保存用システムディスク」をドライブBから取り出す。

```
config.sysとautoexec.batをコピーします。
もとの内容は*.orgとして保存します。

もし都合が悪ければここで CTRL-C （コントロールキーを押しながら “c”
または “C” ）を押して処理を中止してください.
------------------------------------------------------------------------
準備ができたらどれかキーを押してください.
```

```
終了しました。
[RESET]ボタンを押して立ち上げ直してください。
------------------------------------------------------------------------
A>
```

●プログラムの起動

① POWER（電源）を入れる（電源が入っている場合は [RESET] ボタンを押す）。

※ MS-DOSのコマンドラインで mode [⏎] と入力すればMODEが起動しますが、この操作は3章をご覧になってから行ってください。

※添付のインストールプログラムでは、もとからあったCONFIG.SYS、AUTOEXEC.BATをそれぞれCONFIG.ORG、AUTOEXEC.ORGというファイル名で保存して、本システムに必要なCONFIG.SYSとAUTOEXEC.BATをコピーします。
ご自分で設定される方は、「MODEの環境設定」を参考にしてください。

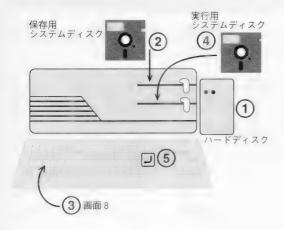
▶▶ハードディスクから立ち上げ、フロッピーディスクに実行用ディスクを作成

●システム用ディスクの初期化

「フロッピーディスクのみのシステム構成で実行用ディスクを作成する場合」の「システム用ディスクの初期化」の一連の作業をしてください。

●データ用ディスクの初期化

「フロッピーディスクのみのシステム構成で実行用ディスクを作成する場合」の「データ用ディスクの初期化」の一連の作業をしてください。

●実行用システムディスクを作成

①ハードディスクドライブから MS-DOS を立ち上げる。

②本書添付の「保存用システムディスク」をいずれかのフロッピーディスクドライブに入れる。

ここでは、ハードディスクドライブ(ドライブ A)から MS-DOS を立ち上げて、ドライブ B に「保存用システムディスク」を入れて作業を進める例で説明します。

③キーボードから以下のように入力。

以下バスマウスの場合で説明します。シリアルマウスの場合は、最初のキーボード入力が違う(6 [↵] と入力)だけです。

```
A>3 c:
```

画面 8　キーボード入力(3.BAT)

※ドライブ C に実行用ディスクを作成する場合、ドライブ名は省略できます。

```
実行用システムディスクを作成します。

「実行用システムディスク」のラベルを貼ってあるディスクを
d:ドライブに入れてください。

もし都合が悪ければここで CTRL-C (コントロールキーを押しながら“c”
または“C”)を押して処理を中止してください。

--------------------------------------------------------------------
準備ができたらどれかキーを押してください.
```

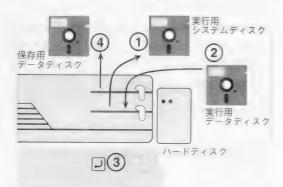
画面 9　3.BAT

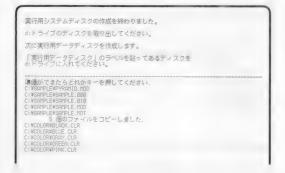
④「実行用システムディスク」のラベルを貼ったディスクをドライブ C に入れる。

⑤ [↵] キーを押す。作業を中止したい場合は [STOP] キー(または [CTRL] ＋ [C])を押す。

```
実行用システムディスクを作成します。

「実行用システムディスク」のラベルを貼ってあるディスクを
d:ドライブに入れてください。

もし都合が悪ければここで CTRL-C (コントロールキーを押しながら“c”
または“C”)を押して処理を中止してください。

--------------------------------------------------------------------
準備ができたらどれかキーを押してください.

        1 個のファイルをコピーしました.
        1 個のファイルをコピーしました.
        1 個のファイルをコピーしました.
C:¥BIN¥CEDIT.EXE
C:¥BIN¥CLS3.EXE
C:¥BIN¥CONE.EXE
C:¥BIN¥CUBE.EXE
C:¥BIN¥DITHER.EXE
C:¥BIN¥DRAW.EXE
C:¥BIN¥GLOAD.EXE
C:¥BIN¥GSAVE.EXE
```

```
実行用システムディスクの作成を終わりました。

d:ドライブのディスクを取り出してください。

次に実行用データディスクを作成します。

「実行用データディスク」のラベルを貼ってあるディスクを
d:ドライブに入れてください。

--------------------------------------------------------------------
準備ができたらどれかキーを押してください.
```

●実行用データディスクの作成

①ドライブＣのディスクを取り出す。
②ドライブＣに「実行用データディスク」のラベルを貼ったディスクを入れる。
③[↵]キーを押す。

```
実行用システムディスクの作成を終わりました。
d:ドライブのディスクを取り出してください。
次に実行用データディスクを作成します。
「実行用データディスク」のラベルを貼ってあるディスクを
d:ドライブに入れてください。
--------------------------------------------------------------------------------
準備ができたらどれかキーを押してください.
C:¥SAMPLE¥PYRAMID.MOD
C:¥SAMPLE¥SAMPLE.000
C:¥SAMPLE¥SAMPLE.010
C:¥SAMPLE¥SAMPLE.MOD
C:¥SAMPLE¥SAMPLE.MOT
        5 個のファイルをコピーしました.
C:¥COLOR¥BLACK.CLR
C:¥COLOR¥BLUE.CLR
C:¥COLOR¥GRAY.CLR
C:¥COLOR¥GREEN.CLR
C:¥COLOR¥PINK.CLR
```

```
実行用データディスクの作成を終わりました。
「保存用システムディスク」を取り出して、「実行用システムディスク」
を入れてください。
[RESET]ボタンを押して立ち上げ直してください。
--------------------------------------------------------------------------------
A>
```

④「保存用システムディスク」をドライブＢから取り出す。

●プログラムの起動

①ドライブＢに「実行用システムディスク」、ドライブＣに「実行用データディスク」を入れ、[RESET]ボタンを押す。

※ MS-DOS のコマンドラインで mode[↵]と入力すれば MODE が起動しますが、この操作は３章をご覧になってから行ってください。

ディスクのバックアップ

「保存用システムディスク」、「保存用ソースディスク」はかならずバックアップしておきましょう。

以下フロッピーディスクのみのシステム構成でバックアップする場合について説明します。

①MS-DOS システムディスク(DISKCOPY.EXE か DISKCOPY.COM の DISKCOPY コマンドが入っているもの)をドライブ A に入れる。
②POWER(電源)スイッチを入れる。
③MS-DOS が立ち上がったことを確認。
④次のように入力。

A>diskcopy　a：　b：⏎

⑤ドライブ A のディスクを取り出す。
⑥ドライブ A に「保存用システムディスク」を入れる。
⑦初期化済みのディスクをドライブ B に入れる。
⑧⏎キーを押す。

ディスクの構成

各ディスクは次のような内容になっています。

▶▶保存用システムディスク

「保存用システムディスク」にはプログラムの実行ファイルやデータファイルが入っています。

●作品例ファイル

作品例が次の 5 つのディレクトリにあります。

ディレクトリ	内　容
¥SAMPLE	本文で使うサンプルファイル
¥GLASS	グラス
¥ROBOT	ロボット
¥TREE	木
¥PICTURE	口絵作品のデータファイル

●バッチファイル

インストール用のバッチファイルがルートディレクトリ(¥)にあります。

ファイル	内　容
INST.BAT	インストールメニュー
1.BAT	フロッピーディスクに実行用ディスクを作成(バスマウス)
4.BAT	フロッピーディスクに実行用ディスクを作成(シリアルマウス)
2.BAT	ハードディスクに実行用ディスクを作成(バスマウス)
5.BAT	ハードディスクに実行用ディスクを作成(シリアルマウス)
3.BAT	ハードディスクから立ち上げ、フロッピーディスクに実行用ディスクを作成(バスマウス)
6.BAT	ハードディスクから立ち上げ、フロッピーディスクに実行用ディスクを作成(シリアルマウス)

●環境設定ファイル

環境設定ファイルは¥SYS ディレクトリにあります。

ファイル	内　容
CONFIG.FDB	マウスドライバ等の設定(フロッピーディスク、バスマウス用)
CONFIG.FDS	マウスドライバ等の設定(フロッピーディスク、シリアルマウス用)
AUTOEXEC.FDD	環境変数の設定(フロッピーディスク用)
CONFIG.HDB	マウスドライバ等の設定(ハードディスク、バスマウス用)
CONFIG.HDS	マウスドライバ等の設定(ハードディスク、シリアルマウス用)
AUTOEXEC.HDD	環境変数の設定(ハードディスク用)
MOUSE.SYS	アスキー社製マウスドライバ

● 実行ファイル

MODE や各ツールの実行ファイルは¥BIN ディレクトリにあります。

ファイル	内　容
CEDIT.EXE	色情報の生成
CONE.EXE	円錐のポリゴンデータを生成
CUBE.EXE	立方体の角をカットしたポリゴンデータを生成
SPHERE.EXE	球のポリゴンデータを生成
POLHD.EXE	正多面体のポリゴンデータを生成
TUBE.EXE	円柱のポリゴンデータを生成
ROT.EXE	ポリゴンデータをもとに回転体を作成
THICK.EXE	ポリゴンデータに厚みを付加
DRAW.EXE	イメージファイルを生成
DITHER.EXE	RENDER が出力したイメージファイルを画面に表示
CLS3.EXE	グラフィック画面の消去
GSAVE.EXE	画面上のイメージファイルのセーブ
GLOAD.EXE	画面上のイメージファイルのロード
MODE.EXE	モデルファイルとモーションファイルの作成
MODE.TXT	ツリーエディタで使えるツールの説明
MODEL.EXE	モデルファイルから RENDER 用のデータを生成
MOTION.EXE	モーションファイルからパラメータファイルを作成
PLANE.EXE	選択点の順番でポリゴンを生成
RENDER.EXE	MODEL.EXE が出力したデータからイメージファイルを生成

● 色指定ファイル

背景色や物体色の指定に使うファイルは¥COLOR ディレクトリにあります。

ファイル	内　容	ファイル	内　容
YELLOW.CLR	背景色(黄)	SHADE.31	物体色(緑)
GREEN.CLR	背景色(緑)	SHADE.32	物体色(水色)
SKYBLUE.CLR	背景色(水色)	SHADE.33	物体色(青)
BLUE.CLR	背景色(青)	SHADE.34	物体色(紫)
PURPLE.CLR	背景色(紫)	SHADE.35	物体色(ピンク)
PINK.CLR	背景色(ピンク)	SHADE.36	物体色(赤)
RED.CLR	背景色(赤)	SHADE.37	物体色(白)
WHITE.CLR	背景色(白)	SHADE.38	物体色(灰色)
GRAY.CLR	背景色(灰色)	SHADE.39	物体色(黒)
BLACK.CLR	背景色(黒)	MAP.PIC	ソリッドテクスチャのデータ(64×64ドットのイメージファイル)
SHADE.30	物体色(黄)		

●部品ファイル

作品例とは別にアルファベットのモデルファイルが¥ALPHA ディレクトリにあります。

ファイル	内　容	ファイル	内　容
ALPHA−C.MOD	アルファベット大文字一覧	ALPHA−S.MOD	アルファベット小文字一覧
A−C.MOD	大文字 A	A−S.MOD	小文字 A
B−C.MOD	大文字 B	B−S.MOD	小文字 B
C−C.MOD	大文字 C	C−S.MOD	小文字 C
D−C.MOD	大文字 D	D−S.MOD	小文字 D
E−C.MOD	大文字 E	E−S.MOD	小文字 E
F−C.MOD	大文字 F	F−S.MOD	小文字 F
G−C.MOD	大文字 G	G−S.MOD	小文字 G
H−C.MOD	大文字 H	H−S.MOD	小文字 H
I−C.MOD	大文字 I	I−S.MOD	小文字 I
J−C.MOD	大文字 J	J−S.MOD	小文字 J
K−C.MOD	大文字 K	K−S.MOD	小文字 K
L−C.MOD	大文字 L	L−S.MOD	小文字 L
M−C.MOD	大文字 M	M−S.MOD	小文字 M
N−C.MOD	大文字 N	N−S.MOD	小文字 N
O−C.MOD	大文字 O	O−S.MOD	小文字 O
P−C.MOD	大文字 P	P−S.MOD	小文字 P
Q−C.MOD	大文字 Q	Q−S.MOD	小文字 Q
R−C.MOD	大文字 R	R−S.MOD	小文字 R
S−C.MOD	大文字 S	S−S.MOD	小文字 S
T−C.MOD	大文字 T	T−S.MOD	小文字 T
U−C.MOD	大文字 U	U−S.MOD	小文字 U
V−C.MOD	大文字 V	V−S.MOD	小文字 V
W−C.MOD	大文字 W	W−S.MOD	小文字 W
X−C.MOD	大文字 X	X−S.MOD	小文字 X
Y−C.MOD	大文字 Y	Y−S.MOD	小文字 Y
Z−C.MOD	大文字 Z	Z−S.MOD	小文字 Z

▶▸保存用ソースディスク

「保存用ソースディスク」にはプログラムソースが入っています。詳しい内容とコンパイル方法については「Appendix B　ソースファイルのコンパイル」を参照してください。

●ディレクトリ

「保存用ソースディスク」には、各コマンドをコンパイルするときに必要なファイルがそれぞれ別のディレクトリに入っています。

ディレクトリ	内　容
¥CEDIT	CEDIT コマンド
¥CONE	CONE コマンド
¥CUBE	CUBE コマンド
¥DITHER	DITHER コマンド
¥DRAW	DRAW コマンド
¥CLS3	CLS3 コマンド
¥GLOAD	GLOAD コマンド
¥GSAVE	GSAVE コマンド
¥LGBIOS	グラフィックライブラリ〝LGBIOS〞
¥MODE	MODE コマンド
¥MODE¥BITMAPS	MODE コマンド
¥MODEL	MODEL コマンド
¥MOTION	MOTION コマンド
¥PLANE	PLANE コマンド
¥POLHD	POLHD コマンド
¥RENDER	RENDER コマンド
¥ROT	ROT コマンド
¥SPHERE	SPHERE コマンド
¥THICK	THICK コマンド
¥TUBE	TUBE コマンド

▶▸実行用ディスクの内容

「実行用システムディスク」、「実行用データディスク」は次のような内容で作成されます。

●実行用システムディスク

COMMAND.COM
AUTOEXEC.BAT
CONFIG.SYS
MOUSE.SYS
CEDIT.EXE
CLS3.EXE
CONE.EXE
CUBE.EXE
DITHER.EXE
DRAW.EXE
GLOAD.EXE
GSAVE.EXE
MODE.EXE
MODE.TXT
MODEL.EXE
MOTION.EXE
PLANE.EXE
POLHD.EXE
RENDER.EXE
ROT.EXE
SPHERE.EXE
THICK.EXE
TUBE.EXE

●実行用データディスク

SAMPLE.000
SAMPLE.010
CUBE.MOD
FLOOR.MOD
FLOORSUB.MOD
PYRAMID.MOD
SAMPLE.MOD
SPHERE.MOD
SAMPLE.MOT
SHADE.30
SHADE.31
SHADE.32
SHADE.33
SHADE.34
SHADE.35
SHADE.36
SHADE.37
SHADE.38
SHADE.39
BLACK.CLR
BLUE.CLR
GRAY.CLR
GREEN.CLR
PINK.CLR
PURPLE.CLR
RED.CLR
SKYBLUE.CLR
WHITE.CLR
YELLOW.CLR
MAP.PIC

▶▸MODEの環境設定

添付のインストールプログラムではCONFIG.SYSとAUTOEXEC.BATを以下の内容で設定します。

●フロッピーディスクにインストールした場合

以下はバスマウスの場合です。

CONFIG.SYS

```
FILES=20
BUFFERS=10
SHELL=A:¥COMMAND.COM A:¥ /P
DEVICE=A:¥MOUSE.SYS B
```

AUTOEXEC.BAT

```
ECHO OFF
VERIFY ON
PATH=A:¥
B:
```

●ハードディスクにインストールした場合

以下はシリアルマウスの場合です。

CONFIG.SYS

```
FILES=20
BUFFERS=10
SHELL=A:¥COMMAND.COM A:¥ /P
DEVICE=A:¥MODE¥MOUSE.SYS S
```

AUTOEXEC.BAT

```
ECHO OFF
VERIFY ON
PATH=A:¥;A:¥MODE
CD ¥MODE¥DATA
```

●マウスドライバ

ご自分で設定される場合は、かならず添付のマウスドライバ(MOUSE.SYS)をご使用ください。

バスマウスをご使用の場合は「MOUSE.SYS B」、シリアルマウスをご使用の場合は「MOUSE.SYS S」と指定してください。

●フリーエリア

MODEの実行にはメモリのフリーエリアが380Kバイト以上必要です。他のデバイスドライバを組み込むときはメモリのフリーエリアにご注意ください。

● MODEの実行に必要なファイル

MODEをご使用になるには添付のマウスドライバのほか、以下のファイルが必要です。

```
DRAW.EXE
DITHER.EXE
GSAVE.EXE
GLOAD.EXE
MODE.EXE
MODE.TXT
MODEL.EXE
RENDER.EXE
```

Part 1
CGの基礎

CGを始めるに当たって、
絵画との違いやCGデザインの心得、
物体の組み立て方について解説します。

1章
数値と感性によるCGデザイン

この章では絵画とCGの違い、MODEシステムを使用する際のCGデザインの基礎、またCGの色彩について概説します。

1.1 CGデザインの心得

私たちはCGに対して作品がこうでなければならないという固定観念にとらわれすぎてはいないでしょうか。特に3次元CGは手法上の制約が多いと思われますが、もっと自由に表現できないだろうかと感じている人も多いと思います。

しかし、CGにはCGの決められた手順を踏まなくては不可能なことがたくさんあります。コンピュータについての知識やその操作も知らなくてはならないでしょう。また、CGの作品を制作する上ではいくつかの選択をしなければなりません。入力の方法、出力の方法、表現方法などがそれです。入力は3D(3次元)や2D(2次元)、出力は画面(カラーテレビと同じように1ラインずつ走査して表示するラスター型とベクトルの合成によって線図形のみを表示するベクトル型に大別される)やプロッタ、プリンタ、フィルムなど、そして、表現の方法としては静止画、アニメーションなどがあります。しかし、これは単なる手法の選択ではなく、広がりゆく創造の世界への選択なのです。

従来、人間は3次元空間の生活の中のものを2次元、つまり平面に置き換える努力をしてきましたが、その代わり描かれた物体の隠れた部分を見ることはできませんでした。その点、3次元CGはいったんデータを入力すれば、さまざまな方向から見ることができ、物体の隠れた面も見ることができます。このように、入出力、表現方法だけを見てもこんなに選択の幅のあるものは今まであったでしょうか。手順を覚えるのは従来の道具ほど簡単ではありません。それなりの努力は当然です。しかし、それを乗り越えれば、もうCGの世界は開かれるのです。

"結果"とともに大切な"経過"

CG はリアルな画像や幾何学的な画像を作る場合に非常に便利なので、そういった表現をしたい人は興味を持つしょう。しかし、CG を使うためにはその技術を理解し、手順を覚えなければなりません。CG ではしっかりした企画書を用意することが創作時のポイントとなります。そして、どのように表現するかの手順が重要です。また、必要な作業をここで明確にしなくてはなりません。特に 3 次元 CG はデータ入力によって作成してゆきますから、しっかりとしたデータを作って手間がかからないようにするほうが近道です。そうして、初めてコンピュータ上でモデリング(modeling)します。モデルが完成したら、物体がどのように見えるかを計算してスクリーンに表示し、色や光の当たり具合などが初めて確かめられます。これをレンダリング(rendering)と呼びます。

CG では物体やそれに対するさまざまな環境を単純なデータに置き換えてゆく作業をモデリングといいますが、具体的なモデリングには物体の形状についての形状モデル、色や質感についての表面モデル、光源の位置や種別についての光源モデル、視点の位置、方向、視野の角度についての視野モデルなどがあげられます。

また、レンダリングは隠面消去(hidden surface removal、3 次元図形の隠れた面を消すこと)とシェーディング(shading、画素に付ける色を決定するためのアルゴリズム)の 2 段階に分けて処理されます。ここまでくると、人間は出力された画像を見て判断をすればよいということになります。

しかし、よりよい作品作りをするためには何度となくチェックを重ね、仕上げる必要が出てくるでしょう。絵画と CG を比較すると作業上の大きな違いがあることが分かってくるはずです。絵画では常に描く人がキャンバスに向かっていなければなりませんでした。しかし CG の場合はデータ入力後はコンピュータが計算して描いてくれます。こういわれるとまったく別のことをするように思うかもしれませんが、人間がイメージを持ってそれを表現するという点ではなんら変わりはないのです。

それでは美術館の展示室に隣り合わせで並べた場合は、どう感じられるでしょうか。過去「CG は芸術になりえない」という言葉を聞きましたが、その言葉は今までの芸術表現との違いやコンピュータというものからくる硬質なイメージが CG への批判につながってきているような気がします。表現方法が大きく違うだけで果たして芸術であるとか芸術ではないとかいえるのでしょうか。また、従来の画家たちには数学などが苦手だという人が多かったようです。ですから、CG の画像データを見ると、何やらわけのわからない数字がたくさん出てきて、どうも嫌だという人も多いでしょう。そういう見方をすると、CG は作曲家が音符という言語を用いて作曲し、演奏されて初めて評価されるといった行程をとるように、美術よりも表現方法が音楽に近いのかもしれません。それは芸術的要素も科学的要素も含む新しい分野です。そしてテレビゲームなどの娯楽、デザイン、映画や CM などの

アニメーション、学校教育、画像処理、宇宙工学などのシミュレーションなど、非常に広い分野で使われています。さまざまな分野で発展していく——それがCGなのです。

感覚や感性を"数値"で表現

コンピュータがこの世に出現したとき、だれがその巨大なもので絵が描けると思ったでしょうか。CGは絵画と違って、すべてが数値で成り立ちます。紙に絵を描くときは、「ここがなんとなく変だな」と思えば色や形などをその場で手直しできますが、CGではそのなんとなくも数値で表現するのです。これは、アニメーションをするときの動きのチェックや時間、またデジタルで音楽を付けるのであれば、その音についてもいえます。

従来の「なんとなく」を数値で表現することは最初はむずかしいでしょう。しかし、これによって表現をする上で人間は何が必要なのかということが確立されてきたように思われます。

造形やデザインの基本的要素の必要性

数値で表現するということがいったいどういうことなのかが少しずつお分かりいただけたのではないかと思います。

しかし、それが理解できたというだけで良い作品が作れるわけではありません。良い作品とは、自分の作りたいものが表現でき、さらに何かを訴えているものです。ただ単にCRT上に表示された画像が人を感動させるでしょうか。CGを手がける人は、それを常に心に留めておいてください。

造形感覚とか、デザイン感覚とかは一朝一夕で身に付くものではありませんが、まず3次元CGで画像を制作する上での基本や、何が重要かだけでも知っておきましょう。

1.2 空間の把握

先人たちは3次元空間の中のさまざまなものを、空間感を意識して2次元の中で表現するためにいろいろな苦労をしてきました。ところが、いざ CG で作品を作ろうとすると、今までの絵画と表現の仕方が違うため、CG の〝空間〟にとまどうことがあります。しかし、現在実際に生活している空間や立体物を構成しているものとほとんど変わりがないと考えてください。何が手前にあるのか、何が奥に位置されているのか、それは平行に置かれているのか、というようなことなのです。

拡大と縮小 —— 物体の大きさの理解

たとえばビルを描く場合、隣に置いた人物が大きければ、高いビルを描いたつもりがそうは見えません。このように物体の大きさは、ある環境に置かれることによって比較の上で大きいとか小さいといいますが、大きさの違いの表現はとてもむずかしいものです。しかし、それを理解できないことには表現も不可能なのです。さて、3次元 CG の場合は、物体を大きくしたり、小さくしたりするのも数値です。それも X 軸上、Y 軸上、Z 軸上にどのくらい大きくするのか、小さくするのかでその大きさを決めてゆきます。

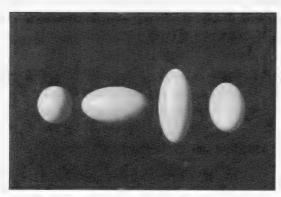



図 1-1　拡大と縮小(球を X 軸上、Y 軸上、Z 軸上にそれぞれ 2 倍)

移動と回転――物体との距離の確認

物体の拡大、縮小の感覚が理解できたら、次は距離の座標表現です。

移動量をどれくらいにすればどのあたりの位置に移動するのかということは、慣れないとすぐには表現できません。特にX、Yの移動は左右、上下の移動となりますが、Zの移動では＋－方向がしっくりこない場合も出てくるでしょう。

たとえば、移動量が大きすぎて画面のはるかかなたへ飛んで行ってしまった、などということは初めての人にはありがちです。しかし、これは作品をたくさん作ってゆけば次第に慣れてきます。

また、3次元CGのよいところに物体を回転できることがあります。回転させていろいろな位置から見て、よりよい構図を選べます。ですから、正確にデータの入力をしましょう。正面から見たときにこれで良しとしても、後方の位置から見たら物が机から浮いていたとか、机の中に物がもぐってしまったなどということはありがちです。回転させる場合もX、Y、Zの数値の入力で行います。このように拡大、縮小、移動、回転を数値で入力し、表すことによって形を作ったり空間をとらえたりしていくのです(口絵①参照)。

三面図

3次元の世界を扱う場合には、その中にある形状や位置がX、Y、Z座標値で扱われることが理解できたと思いますが、その座標値データの正面図、平面図と側面図がしっかりと把握できれば、物の形状もよく分かります。その反対もいえます。実行例については後述します。

1.3　空間を表現

3 次元 CG は実際に計算によって 3 次元の表現をしていますが、座標による空間感の把握ができれば、慣れるにしたがって手に取るように理解できるようになります。しかし、それだけでは十分ではありません。座標系のほかに構成(配置)、色、遠近や光などの視覚的な空間要素を加えれば、さらにリアルな空間感も表現できます。

2Dと3Dの空間表現

2D の表現とは X(横方向)と Y(縦方向)のいわゆる 2 次元の情報しか持っていないものをいいますが、3D は X、Y のほかに Z(奥行き)の情報を持つものをいいます。3 次元 CG は見ているディスプレイ上では 2 次元ですが、実際には 3 次元のデータとして成り立っているので、初めての人には少し不思議な感じがするかもしれません。また X、Y の考え方は比較的簡単ですが、それに Z が加わり、さらに移動回転させたりすると頭の中が混乱する人もいるでしょう。このように 2D と 3D では大きな違いがあります。

このように 3D での空間表現は Z の情報によってとらえられますが、2D の場合の空間表現には従来と同様にさまざまな技を使わなくてはならないのです。たとえば古来から絵画の中で使用された遠近法などもそうですし、形に大小の差を付け、空間的に隔たりを感じさせたり、あるいはオレンジ色などの暖色系の色は前へ向かって進出する色、ブルーなどの寒色系は後退する色などという色の性質を使って表す方法もあります。

つまり 2D は描く人のデッサン力や感性がそのまま絵として表れますので、その人の力量が大いにものをいいます。しかし 3D では 3 次元のデータで作っていくので、物体を Z 軸上に遠く離せば、パース(透視図)によって自然にその物体は小さくなります。3D はパースが自動的に付くわけですが、さらにアニメーションによって空間の把握は容易になります。

このように 2D と 3D では制作過程において共通でないことが多いと思われます。しかし、美しいものを作るということでは共通ですし、2D の遠近法をシェーディングに生かすなどの工夫をすれば、さらに空間感が得られるでしょう。

空間感を表現する構図

単に空間を表現するといってもいろいろな表現方法があります。空間自体が狭く感じられるか、広く感じられるか、遠近感があるのかないのか、それは構図によってかなり違います。図 1-2 を参考にしてみてください

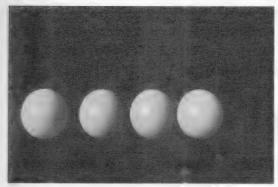
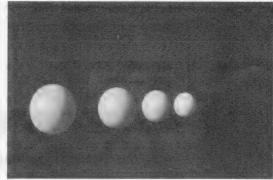



図 1-2　空間感の表現

遠近感の表現

1 点透視法的な見方で、空間感のある構成をしてみましょう。物が消失する点を決め、消失点に BOX が吸い込まれていくように、消失点に近いものほど Z の移動量を大きくします。移動、回転などを加え、構成に変化を付けてみます。しかし、これは 3 次元 CG の 3 次元データによる表現ですから、本当に BOX は遠くにあるものほど遠くにあるという感じが表現されています。しかし、2 次元の場合は遠くにあるものを小さくしてその変化を表現しなくてはなりません。これが大きな違いなのです(口絵②参照)。

空間を表す色彩

同じ大きさの BOX を移動させ、回転を加えるだけで奥行き感が出ましたが、まだなんとなくしっくりきません。たとえば色彩を使ってみるのも 1 つの手です。ここではバックの色に黒に近い青を使い、遠くにある物体ほどバックの色に近づけて、溶け込ませるような効果をねらっています(口絵③参照)。

光の明暗の利用

太陽や電気などの光があるからこそ、私たちには物が見えます。あまりにも当たり前すぎて、私たちは光について普段から意識が足りないようです。3 次元 CG を経験した人は光の重要性を非常に身にしみて感じるものです。

MODE の光源を大別すると、太陽の光のようにどこにもまんべんなく照らす平行光源、そして輝度の変化が非常に激しくある部分を照らすのに便利な点光源の 2 つがあります。空間感を出す構成ということでは手前の物体は光を多く受け、遠くに行くほど序々に光を受ける量が減少するほうがよりよいと思われます。平行光源を使った場合と点光源を使った場合の感じをよく見比べてみてください(口絵④⑤参照)。

1.4　構成の基本的要素

さて、作品を見たときによく「構成力がある」、「構成力がない」といいますが、ではその構成力とは一体何なのでしょう。そして、どうすれば構成力が身に付くのでしょうか。紙の上にただ何となく絵を描いていても構成の勉強にはなりません。

構成は点、線、面、立体の基本的要素とその配置、分割、組み合わせなどのコンポジションによって成り立ちます。まず、この基本的要素の点、線、面、立体ですが、点には本来大きさはありませんが、解像度の関係でコンピュータのディスプレイ上ではピクセルの大きさ程度で表現されます。線は点がつながったものや移動してできた軌跡です。面はさらに縦横に広がったり、回転したり、移動したりしてできるものであると考えればいいのです。立体は面をさらに移動、回転を加えることによって形作られるものです。

点

構成の要素には形を作るものの１つとして点があります。点には幾何学上の位置は存在しますが、大きさはありません。しかし、デザイン上は大きさや色を持って示されます。形は円で表されることが多く、位置と力を表します。たとえば空間の中の１点は非常に集中力が集まりますが、２点になりますと集中力も２分の１に減ります。

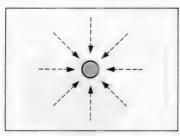
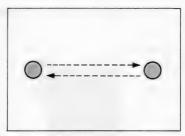



図 1-3　点の構成

線

線は点が移動したもので、デザイン上は幅や長さ、色を持っています。線には直線と曲線がありますが､1本でも方向性や動きが感じられます。また直線を平行に引いたときと交差させたとき曲線を使っただけでも画面の感じが変わります。そしてなんとなく躍動感が出たり柔らかく感じられたりします(口絵⑥参照)。

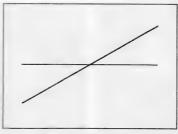
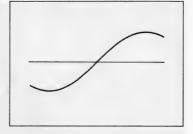
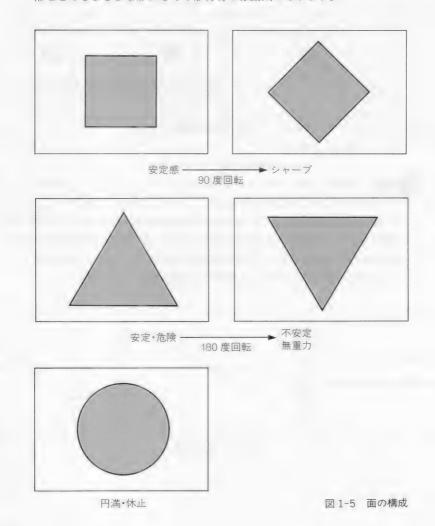



図 1-4　線の構成

面

面はその形だけで意味あいを持ち、1つの塊を感じさせます。また、その形にはさまざまな色で塗られることによって、意味あいが変化します。たとえば正方形は堂々として安定感がありますが、90度傾けるとシャープな感じになります。三角形は安定感はありますが危険な感じも受け、逆さにすると不安定になり無重力な感じも受けます。円形はどう傾けても円形ですが、円満で休止している形です。これらは基本的な形ですが、長方形や楕円形などのさまざまな形にもその形特有の雰囲気があります。

図1-5　面の構成

立体

立体は面の軌跡、積み重ね、回転によってできます。立体とは表面が平面や曲面で構成された3次元物体のことで、その基本形には球体、円錐、立方体、円柱、三角錐、角柱があります。これは3次元CGの持つ基本形態とほとんど変わりません。また、立体のうち平面のみによるものを多面体といい、正多面体の中には5種類の正多角形を2種類以上で組み合わせた準正多面体などがあります。

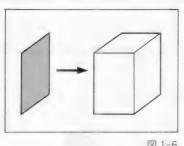
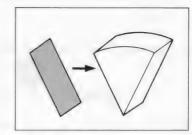



図1-6　面の軌跡

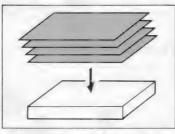



図1-7　積み重ね

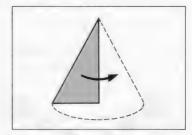



図1-8　回転

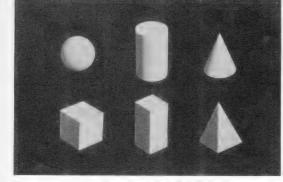



図1-9　立体の基本形

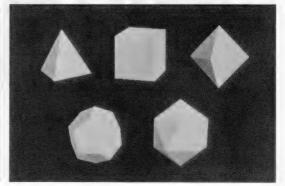



図1-10　正多面体の種類

配置

点、線、面などの基本的要素が揃っても、構成ができるとは限りません。それらの組み合わせによって初めて構成が生まれます。その1つに配置があり、たとえば円形だけでも配置によって上昇した感じや下降する感じが表現できたり、リズム感が出たりもします。また左右対称のシンメトリの構図は動きに乏しいのですが､安定性があり均整がとれます。その反対のアシンメトリは偏った配置によってバランスを崩すやり方で、動的な感じを受けます。日頃からこれらのさまざまなパターンを知っておくと作品制作のときに大いに役立ちます。

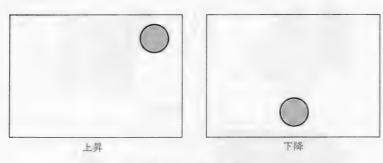



図 1-11　上昇と下降

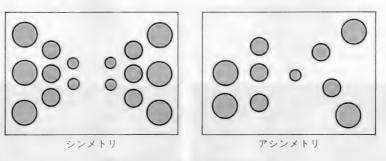



図 1-12　シンメトリとアシンメトリ

分割

表現のスペースは1枚の紙ならばその面積の中で考えるわけですが、その面積を分割する方法はいろいろあります。直線でも曲線でも分けられますし、等分割にする場合でも形は異なっても面積は等しい等量分割と形も面積も等しい等形分割があります。また画面を自由に分割する自由分割は、数学的な規則を用いると比較的美しい分割になります。最も一般的な分割はギリシャの昔から研究されてきた黄金分割で、縦横比1：1.618です。

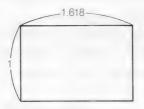



図1-13　1：1.618黄金矩形

組み合わせ

形の組み合わせには、離れる、接する、重なるなど約47通りあるといわれています。図のような円形と正方形の組み合わせに、それぞれの形を塗りつぶすことによって形のイメージも変わります。

離れる

接する

重なる

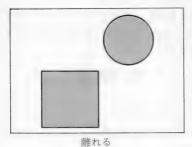
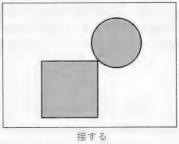
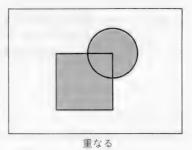



図1-14　形の組み合わせ

構成を考えるときには、これまでに説明してきたようなさまざまな要素を考慮しましょう。優れた作品でも構成自体はそれほど凝っているわけではないので、日頃からいろいろと試してみてください。

さまざまなポリゴンによる構成

さまざまな幾何形態を用い、移動、回転を使って簡単な構成を行ってみます。使用する球、直方体、円柱、円錐といった形は形態の基本となるものであるとともに、私たちの生活の中で使われる物のほとんどがこれらの形態の組み合わせによっています（口絵⑦⑧参照）。

構成といえば私たちがテーブルの上に皿、コップ、ナイフ、フォークなどの食器類を置くことも構成の１つといえるかもしれません。このように日常の中で与えられた形を構成することも大いに勉強になるでしょう。

次に幾何形態の構成によって、泣いた顔、笑った顔、怒った顔の表現をしてみましょう。幾何形態の構成というと硬質な感じがしますが、こんな楽しい表現もできるのです（口絵⑨⑩⑪参照）。

1.5 色　彩

私たちはふだん何気なく色を見ていますが、色はどのようにして目に入ってくるのでしょうか。たとえば１本の赤いバラは、太陽の白色光がこのバラを照らすとき、表の色素特有の成分が赤以外の色光を吸収し、赤い色のみを反射します。すると、これが目に入り、赤いバラであると知覚されるわけなのです。

色には、赤や黄色といってもさまざまな色があります。オレンジ色に近い赤なのか、くすんだ黄色なのか、また隣合う色や、見る人のその日の体調、そして見る時刻によっても感じ方が違います。このように色は形とは違って非常にあいまいな要素が多く、もっと元気な黄色とか、甘い黄緑などというように表現されることもありますが、CGではそのあいまいさを数値で表さなくてはならないという原則があります。

色彩の三原色

自分で色を作ってみましょう。CG は色がすぐに変えられるという利点はあっても、絵具を扱うときと同じように色を作るためには CG の色の規則を理解する必要があります。

それでは、その CG の色の規則とはどんなものなのでしょう。私たちが生まれてから絵具や色鉛筆などを使って出した色と CG で扱う光の色とでは大きな違いがあります。物体の色は Yellow、Magenta、Cyan、の三原色の混ぜ合わせで作り出します。これを減法混色(減算混合)といい、色を混ぜれば混ぜるほど、黒に近い Gray になります。

また、それと反対に光の色は Red、Green、Blue で成り立ち、この 3 色の混合によって色を作り出します。これを加法混色(加算混合)といい、色を混ぜれば混ぜるほど白に近くなります。一見むずかしい感じがしますが、考え方を少し変えればそれほどむずかしいものではありません(口絵⑫参照)。

色彩の三属性

色には色相、彩度、明度があります。色相は色合いの違いをいいます。口絵⑬の色相環は、24 の色相を紫みの赤を 1 として、24 の赤紫までを配置しています。たとえば、対角線上にある色を並べるとはっきりとして動的な感じがします。隣合う色を並べるとまとまりが出てすっきりとします。しかし 4 つ先の色を並べるとなんとなく落ち着きのない配色になります。このような知識も作品を作る上で重要なことです。

また色にはこのように色相を持つ有彩色と白、黒、灰色などの色相を持たない無彩色があります。無彩色は明るさだけを持ち、これを明度といいます。もちろん有彩色には明度も鮮やかさの度合いの彩度も持っています。色相、彩度、明度を立体的に表すと口絵⑭のようになります。

配　色

作品は使う色によってもかなりイメージが変わってきます。そして配色に気を配ると表現したいイメージがそれなりに見えてくる場合もあるでしょう。口絵⑮の左は 2 色配色ですが、この 2 色だけでもダンディーな感じやすがすがしい感じ、甘い感じなどが表現されます。このように隣合う色によって色というものは違った感じにとられるのです。

また 3 色以上にして面積比を変えてみたり、アクセントを付けたりして、いろいろなイメージを色で表現してみました。イメージによる配色を日頃からよく知っておくと作品制作に役立ちます。

物体色

Red、Green、Blueの色を原色で使うときは色で悩むことはありませんが、微妙な色を作ろうとする場合は話が違ってきます。自分の思い描いていた色が、レンダリング後に全然違っていたということはありませんか。空間と同じように、色についても何度もレンダリングをするよりは思った色に近いものをすぐに出せるほうがよいでしょう。そのためには気に入った色のデータを色見本などでいつでも使えるように作っておくと便利です。また、最近ではRGBの色見本帳も発売されています。

物体の色は、物体の表面から目に届く光によって決定されます。物体の色は、大まかにdiffuse（拡散反射光）、ambient（環境光）、specular（鏡面反射光）の3つの成分に分けられます。

光源色

光源についての詳細は後述しますが、光に色を付けてみると画面全体の感じがどう変わるでしょうか。物体の色に光の色が加わるので、さらに加法混色が行われます。たとえば単なる白色光ではなく、乳白色などの春らしい光の色や柔らかい光の色などを付けることでもかなり画面全体の雰囲気が変化します。基本的な色にそれぞれ赤、黄、青、緑の光を当て、その変化を見たのが口絵⑯です。

1.6 光

光があるから物が見える。あまりに当然過ぎて私たちはなぜ物が見えるのかを考えることはないかもしれません。太陽の光、電気の光があるからこそ物が見え、形も色も分かるのです。またそれと同じように3次元CGに光は欠かすことのできないものです。光をいろいろと設定してみることによって、ものの感じも画面の感じもかなり違って見えます。

平行光源と点光源

光源の種別や強さを変えることによって、物体そのものの質感や雰囲気を変えることができます。

平行光源は太陽のような光で、光の強さが距離に関係なくどこでも一定であるのに対し、点光源は豆電球やランプのように光源自体の大きさが無視できるほどの小さくて近い光源です。点光源の場合は、物体に照射される光の強さは光源から物体までの距離が遠くなるほど弱くなり、強さは距離の2乗に比例して弱くなります。つまり距離が2倍になれば光の強さは4分の1になり、距離が3倍になれば光の強さは9分の1になります。

diffuse(拡散反射光)、ambient(環境光)、specular(鏡面反射光)

「物体色」でも触れたように、光を当てたときの物体の色はdiffuse、ambient、specularで成り立ちます。それぞれ拡散反射光、環境光、鏡面反射光を意味していますが、これらを使っても雰囲気を変えることができるのです。

- diffuse(拡散反射光)　…　物体の表面や内部で乱反射された光
- ambient(環境光)　…　まわりの物体の照り返しや、空気による拡散光(自身が発光体の場合の光も含む)
- specular(鏡面反射光)　…　ハイライトの色(光源の光が物体の表面で反射し、直接目に入ってくる光)

diffuseは物体の表面や内部に入り込んだ光がその中で反射を繰り返し作られるので、まわりの物体の2次反射光となります。ambientはまわりの物体からの照り返しや自らの発光を光源とする陰の色を表します。普通光が当たって、陰となる部分は物体そのものの色と大きな変化はなく暗くなる程度ですが、まったく違う色を付けることによって抽象的なおもしろい変化も得られます。specularは輝きが1番鋭い部分の色の成分で、物体の特徴を決定する上で最も重要となります。金属的な物体ではspecularの色はdiffuseの色と同じ、プラスチック風の物体では白色となります(口絵⑰参照)。

また、それぞれについてRGBの割合で色を作るので、ambientのRGBの割合を変化させることで赤っぽい陰にしたり、specularの強さによって質感を変化させることができるのです。specularを決定する数値としては、RGBの色のほか、表色の粗さを指定するものもあります。

diffuse、ambient、specularの割合を変化させるだけで、陶器風と金属風の壺が表現できます(口絵⑱参照)。

1.7 デザインのまとめ

デザインはその応用によっていろいろに分類されます。たとえばグラフィックデザインやファッションデザイン、プロダクトデザインなどがそれです。しかし、最近では幅広くデザインや造形を扱う仕事をしたいという人が多くなってきているのが実情です。

第1章でデザインの基礎についてお話しましたが、構成や色彩、表現などはたとえ別々に分類されていても、同時に考えなくてはなりません。またデザインは伝達手段として実用的要求を満たすべきものでなくてはなりません。

ここで紹介する作品はデザインよりアート性が強い感じがしますが、一般的にアートとデザインの相違はあまり理解されていないことが多いようです。ものを作るという点で同一視されるためでしょう。しかし、デザイン的に優れたものは1つの美術作品にもなりうるといったところで、あくまでも美的なものが要求されるという共通点はあります。

2章
アニメーション

「アニメーション(animation)」というと、コマーシャルで使われているきらびやかなCG、あるいはテレビ漫画に代表される「セルアニメ」を思い浮かべる方もいると思います。新聞、雑誌などのテレビ番組紹介欄の「アニメ」というジャンルで紹介されている番組のほとんどが劇画、漫画などを原作とする「セルアニメ」です。

ここでは「アニメーション」を語源から掘り下げ、「コンピュータ」とのかかわりについて解説します。

2.1 animationとコンピュータ

アニメーションというと「セルアニメ」といわれるくらい、実に多くの作品がセルアニメの手法で作られ、テレビなどのメディアで紹介されてきました。また、数多くの「きらびやかな物体が空間を飛び回る」CGがコマーシャルフィルムやイベントなどで利用されています。

ではアニメーションとは何でしょうか。アニメーションを言葉の意味から見てみると、animationはanimaから派生した語で、次のような関連語があります。

anima　　　＝＜名詞＞　生命、心、魂
animal　　　＝＜名詞＞　動物
　　　　　　　＜形容詞＞動物性の
animate　　＝＜動詞＞　生命を与える、生かす、元気づける、活気づける、動かす
　　　　　　　＜形容詞＞生きている、生きた、元気のある、活気のある、動く
animation　＝＜名詞＞　生気、活気、生き生きしていること、活気づけ
animator　　＝＜名詞＞　生気を与えるもの

通常「動画」と訳されるanimationは、もともと「生きているものとして、生き生きと動いている状態」を示す言葉なのです。

一般によく使われる「漫画映画」の意味は、animated cartoon（活気に満ちた連続漫画）を短縮してanimationになったようです。これは、たとえば単に「車」（くるま）といった場合、一般に「自動車」を指すのと同様です。

ではどうして、アニメーション＝漫画映画＝セルアニメ、の図式ができたのでしょうか。その流れを簡単にまとめてみましょう。

①生き生きと動きまわるものを画面上に作りだそうとした。つまりアニメート（animate）された絵を描こうとした。
②メディア上の制限から、ある程度の時間や動きを表現するには、数多くの絵を描かなければならない。
③油絵具などで細密に描いていたのでは枚数が稼げないので、表現としてはキャラクターの特徴を抽象化、象徴化、単純化できる漫画の手法を選ぶ。
④画面上の動くところと動かないところを別々に処理し、制作効率を上げるため、透明なセル（セルロイドの略、現在は他の合成樹脂を使用）を用いた手法が開発される。
⑤マスメディアの発達でより多量の作品の安定供給が求められる中で、セルアニメーションは動画制作の代表的なテクニックとなる。

このように作品としての安定した質と量の生産性が優先された結果、アニメーション＝セルアニメ、と受け取られるようになったと考えられます。

素朴なパラパラ漫画やからくり仕掛けの装置から始まったアニメーションは、映画フィルム、ビデオなどの時間変化を記録できる映像メディアの発達で、その表現の可能性を大きく広げています。そして、アニメーションは本来の意味に照らし合わせて、実に多様な形態で存在しているのです。

2.2　時間の概念

コンピュータという大きく発達しつつあるメディアと複合されても混乱しないようにアニメーションの定義をしておきましょう。

● アニメーション＝時間の変化と形態の変化。広い意味で、形態の持つ本来の時間軸を変化させ、新たな時間的変化を持たせた表現形態。実際には、変化する１コマ１コマの絵を動かすことによる、動く絵

アニメーションを本来の意味である「動きを与える」、「動きを与えられたもの」として定義すると、次のようになります。

● アニメーション＝自分が動かそうとしたものが動いたときの感動

動物である人間は、本能的に動くものに対して非常に敏感です。自分に対してなんらかの影響を与えそうなもの、たとえば近づいてくるものには、無意識に注意を払っていると思います。動物として、自分の生命を守るために必要な基本的能力の中に、「時間」と「空間」を感じ、分析する力があるためでしょう。

2.3　アニメーションデザイン

時間軸がないと片づけてしまわれがちな2Dの絵画でも、時間を感じさせる作品は何回見ても新鮮な感動を与えられるものです。描かれた前後の情景が感じとれるものや、音楽的リズムが聴こえてくるようなものなどです。また、その時間も何万年という大地の動きのスケールから、1000分の１秒のほんの瞬間まで、時間を感じさせるものはさまざまです。

これは、典型的な瞬間表現ともいわれる〝写真〟にも逆説的にあてはまります。写真を撮るとき、ファインダーを覗いて構図を決めてシャッターを押す。その瞬間を記録したはずの映像を見てみると、笑っているのか、怒っているのか、分からないときがあります。

つまり人間の表情も大地の情景もその特徴に時間軸を含んでいるのです。時間を感じさせる絵画や写真は、それを集約する凝縮された瞬間をとどめているのでしょう。

アニメーションデザインは、その凝縮された瞬間をも含む複数の画面を時間軸上にいかに展開するかが問われます。また、どういう強調のしかたをするかに感性が現れます。

アニメーションを制作

アニメーションの制作を具体的にみてみましょう。

■何を動かすか

アニメーションは動きを与えてこそ初めて意味を持ちます。動かしたいものは何で、どんな性質のもので、誰に見てもらうのかを、しっかりと決めておきましょう。あいまいな状態で制作を続けると、散漫なアニメーションになりがちです。

動かすものは、大きく分けて２つあります。主体と客体です。

主体はあなた自身、アニメーションを見ている人です。カメラでいえば、その位置、方向、画角などです。アニメーションでは現実では不可能なカメラワークもできます。

客体はそのアニメーションに登場するものすべてです。動くはずのないものも動かせるのがアニメーションの醍醐味です。

アニメーションはコミュニケーションの手段の１つでもあります。無駄のない、かつ豊かな表現のためにも、何をどう動かすかは演出の基本となります。

■動きを分析

動きを表現するためには、まず自分が表現したい動きをよく知る必要があります。日常、当たり前のように見ているすべてのものの動きを〝作り手〟の立場で観察するのです。特に現実に存在する動きを表現するときは、さらに注意深い観察をしましょう。

分析、観察すべきものは、動きの方向、基本リズム、テンポ、抑揚、重心の変化などです。

たとえば、人が歩く様子を観察する場合です。まず腰のあたりの重心の部分に注目し、それがどの方向へどのくらいのテンポで動いているかをつかみます。次に、大きな動きの方向から小さな上下左右の揺れまで、基本リズムを探ります。最初から細かな部分の輪郭を追ったりせずに、骨組みを透かせて、骨格だけの動きで観るのもよいでしょう。逆に部分部分に眼をつけてその動きの軌跡を追うのも表情の分析には必要です。慣れてきたら動きの抑揚を時間軸上で把握できるところまで観察しましょう。〝受け手〟の立場では気がつかなかった動きの軸が見えてくるはずです。

ビデオ機材を利用できるときは、コマ送り機能を活用して細かく観てみるのをお勧めします。正確な時間軸の分析に役立ちます。

架空の物体を架空の空間で動かすアニメーションであっても、物理法則が適用できないような世界の出来事では、見ていて奇妙です。それもアニメーションならではの面白さですが…。しかし〝あるキャラクターの歩行〟の動きを表現して、それが重力の小さい月面での歩きのようになってしまったら、それはそのキャラクターの個性ではなく、その環境を説明することになってしまいます。押さえるべき動きのペースも知っておくとよいでしょう。

適度の強調とさりげない現実感を感性でバランスをとる。動きの分析は第2の基本です。

■方法を選択

動く絵を作るための最も基本的で簡便な方法は、たくさんの絵を描いてパラパラとめくりながら見ることです。一般にパラパラ漫画といわれる手法です。子供の頃、ノートや教科書の余白を使って遊んだ方もいると思います。素朴な手法であるため融通もききます。めくる速さを変えてリズムを変えたり、向きを逆にしてストーリーを反対にしたり。でも、いかんせん大勢の人に一度に見せられませんし、長いストーリーを組み立てるのは困難です。

そこで、そのたくさんの絵を何かに写し撮って連続的に上映できるメディアが登場します。映画フィルム、ビデオです。フィルムは1秒間に24コマ、ビデオは1秒間に約30コマ(日本、米国等で使用されるNTSCフォーマットの場合)の画面を再現できます。これらのメディアに1枚1枚、絵を撮影して録画することを「コマ撮り」といいます。アニメーションの場合は、かならずしも1秒間に24枚、30枚の絵が必要というわけではありません。動きは粗くなりますが、複数コマずつ絵を変えていくこともできます。

フィルムやビデオでのコマ撮りを用いると、1枚1枚の絵は別に〝絵〟でなくてもよくなります。実物を撮影してコマ撮りを行うものです。粘土などで作った人形を少しずつ動かしてのアニメーションも可能になります。また、風景を取り込んで雲や波の動きを時間のスケールを変えて撮影することで、広い意味でのアニメーションの領域を拡大できます。そして、さらにコンピュータによってアニメーションの実現方法は豊かに広がりつつあります。

アニメーションにコンピュータを使用

コンピュータの利用により、アニメーションの表現の可能性は大きく拡大し、多様な分野でなくてはならないものになってきました。コンピュータとアニメーションの接点を説明します。

■キャラクターの保存

CGでは、コンピュータに与える情報で2次元CG、3次元CGと分けています。2次元CGは2次元の、3次元CGは3次元の情報を基に処理しています。2次元CGではコンピュータを絵筆と絵の具、あるいは精密な製図機のように利用できます。それによって、アニメーション用の数多くの画面やキャラクターの線画を描くことができます。セルアニメーションのように変化する部分としない部分を合理的に管理することも可能です。

■キャラクターの生成

CGならではの映像に3次元CGがあります。キャラクターを3次元で形作ることができれば、それを自由な方向に動かしたり、立体物どうしの変形や光源を含む環境の変化のシミュレーションも可能です。

■カメラの動き

3次元CGではあなたの眼の位置、状態を連続的に変化させられます。現実では不可能な視点の動きも、物理法則に基づくリアルな動きもできます。

■時間のコントロール

動きのリズム、テンポ、抑揚などを数量的に扱えるので、数少ない動きの節々から滑らかな動きを補完できます。物理法則に従った動きの表現などはコンピュータの独壇場です。

■出力メディアのコントロール

多数のアニメーション用画像をフィルムやビデオにコマ撮りする作業にコンピュータを利用できます。試行錯誤や同じパターンの繰り返しなどには最適です。

Part 2
MODEシステム

MODEシステムの基本的な使い方について説明します。
まず静止画を作るために
ポリゴンエディタとツリーエディタを説明し、
次に動画を作るためのモーションエディタを説明しています。
そして最後に作品を作ります。

3章
MODEの概要

MODEとはモデリングおよびモーションデータの作成をサポートするツールです。MODEではモデルファイル、キーフレーム、そしてモーションファイルを画面上でシミュレートしながら作成できます。また作成したデータはテキストファイルですから、テキストエディタを使っても編集できます(「Appendix A　テキストエディタを使った編集」参照)。

MODEにはモデリングするための「ポリゴンエディタ」、いくつかのデータを組み合わせて複雑なデータを作るための「ツリーエディタ」、そしてモーションデータを作成するための「モーションエディタ」があります。これらの機能を利用してアニメーション作成までできます。

ポリゴンエディタは、三面図やパース(透視図)を見ながら、視覚的にモデリングするためのエディタです。ポリゴンデータを作成するだけでなく、拡大、回転といったさまざまな編集機能も備えています。

ツリーエディタは、ポリゴン以外のデータ作成やツールを使ったポリゴンの作成を行うためのエディタです。複数のデータを読み込んで編集できます。

モーションエディタは、ポリゴンの動きをデザインするためのエディタです。ツリーエディタで動かす部分を決め、ポリゴンエディタで確認します。

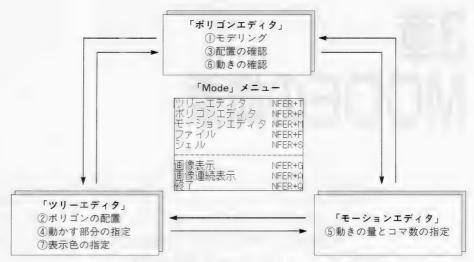
この章では添付のディスクに収録されているサンプルデータを使って、MODEの各機能を説明していきます。

1) アニメーション表示するときは、①～⑦の後に「画像連続表示」を選択
2) 動きのない画像をカラー表示するときは、①、②、③、⑦の後に「画像表示」を選択

図 3-1　MODE システム——3 つのエディタ

3.1　MODEの起動

実際にデータを作る前にサンプルデータを読み込んでみましょう。

①ドライブ A に「実行用システムディスク」、ドライブ B に「実行用データディスク」を入れ、RESET ボタンを押して立ち上げます(ハードディスクにインストールした場合はドライブ B をインストールしたドライブに置き換えてください)。
②次のように MS-DOS のコマンドラインから入力して、MODE を起動します。

```
B>mode　sample↵
```

この場合、次のファイルが読み込まれ、後述するツリーエディタ画面が表示されます。

sample.mod	モデルファイル
sample.mot	モーションファイル
sample.000	0 コマ目のパラメータファイル
sample.010	10 コマ目のパラメータファイル

※モデルファイルは、物体の形状を定義したファイルで、モーションファイルは、パラメータファイルとコマ数の対応を示すファイルです。またパラメータファイルとは、あるコマにおける変数の値を定義したファイルです。

3.2 MODEの終了

何も編集しないで、MODEを終了してみましょう。

①画面左上の「Mode」上でドラッグ（マウスボタンを押したまま移動）してマウスを下に動かし、「終了」の上でマウスボタンを放します。または NFER ＋ Q （NFER と Q を同時に押す）を押します。
この操作は各エディタに共通です。
② MS-DOSのコマンドラインにもどります。

※データを編集した場合、「"というファイルをセーブしますか？(y or n)」とデータをファイルに保存するかどうかを聞いてきますので、次のようにしてください。

1)　保存して終了する場合は、Y キーを押します。
2)　保存せずに終了する場合は、N キーを押します。もう一度確認の意味で「セーブしていないファイルがありますが、終了しますか？(Y or N)」と聞いてきますので、SHIFT ＋ Y を押します。なお終了を取りやめてMODEにもどりたいときは SHIFT ＋ N を押します。

3.3 「Mode」メニュー

それではMODEの各機能を使ってみましょう。再びmode　sample⏎を実行してMODEを起動してください。

MODEはツリーエディタから起動し、画面左上に「Mode」メニューが表示されます。ここへマウスカーソルを移動させてドラッグしてください。そうすると**画面3-1**の「Mode」メニューが表示されます。

ツリーエディタ	NFER+T
ポリゴンエディタ	NFER+P
モーションエディタ	NFER+M
ファイル	NFER+F
シェル	NFER+S
画像表示	NFER+G
画像連続表示	NFER+A
終了	NFER+Q

画面3-1　「Mode」メニュー画面

それぞれを実行するには、ドラッグして実行したい項目までマウスカーソルを移動させ、ボタンを放します。

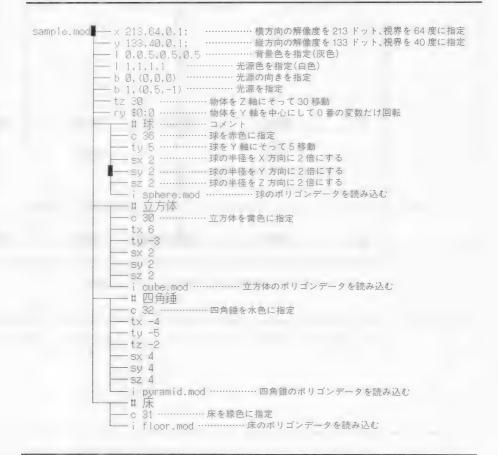
なお、「Mode」メニューにある項目は、キーボードでも選択できます。使用するキーは次のとおりです。

ツリーエディタへ	NFER + T （XFER + T）
ポリゴンエディタへ	NFER + P （XFER + P）
モーションエディタへ	NFER + M （XFER + M）
ファイルの切り替え	NFER + F （XFER + F）
シェルの起動	NFER + S （XFER + S）
画像表示	NFER + G （XFER + G）
画像連続表示	NFER + A （XFER + A）
MODEの終了	NFER + Q （XFER + Q）

ツリーエディタ

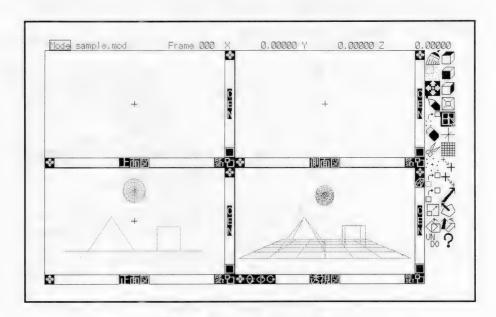
起動直後はツリーエディタが選択されており、画面にはいま読み込んだデータが表示されます。

「sample.mod」のデータにはそれぞれ以下のような意味があります。

画面 3-2　ツリーエディタ画面

ポリゴンエディタ

「ポリゴンエディタ」を選択してみましょう。「Mode」メニューをドラッグして、「ポリゴンエディタ」でボタンを放す（または NFER ＋ P ）と次の画面が表示されます。

画面3-3　「ポリゴンエディタ」画面

それではワイヤフレームを動かしてみましょう。モーションファイルが用意されているので、 スペース キーを押すごとにワイヤフレームがY軸を中心にして動きます。

ファイルの切り替え

「ファイルの切り替え」は、複数のファイルを切り替えながら編集できる機能です。

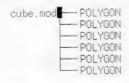
「Mode」メニュー(または NFER + F)で「ファイルの切り替え」を選択してみましょう。

```
ファイル名                                            エラー 編集中
------------------------------------------------      ------ ------
                                                        no     no
sample.mod                                              no     no
sphere.mod                                              no     no
cube.mod                                                no     no
pyramid.mod                                             no     no
floor.mod                                               no     no
floorsub.mod                                            no     no
```

画面 3-4 「ファイルの切り替え」画面

画面には「sample.mod」を始めとして拡張子が「*.mod」のファイルが6行表示されています。

次にこの中の「cube.mod」を選択してみましょう。キーボードから J (下へカーソル移動)を押してカーソルを「cube.mod」に合わせ、 スペース を押します。そうするとツリーエディタにもどり、今度は「cube.mod」の内容が表示されます。

画面 3-5 ツリーエディタ画面(立方体)

この状態でポリゴンエディタを選択すると「cube.mod」の図形だけが表示されます。

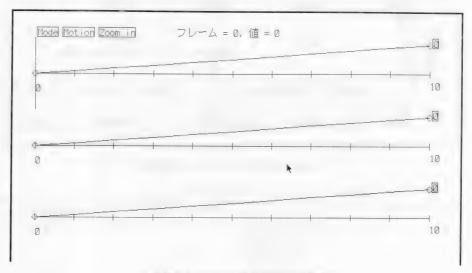
「sample.mod」の画面にもどるときは、再び「Mode」メニューで「ファイルの切り替え」を選択し、 K (上へカーソル移動)、 スペース で「sample.mod」を選択します。

シェルの起動

「Mode」メニュー(またはNFER＋S)で「シェルの起動」を選択すると、いったんMS-DOSのコマンドラインにもどります。次のように入力すると、再びMODEにもどってシェルを起動する前の画面を表示します。

B>exit↵

モーションエディタ

「sample.mod」の画面にもどり、「Mode」メニュー(またはNFER＋M)の「モーションエディタ」を選択すると次の画面が現れます。

画面3-6 「モーションエディタ」画面

ポリゴンエディタで動画表示できたのは、モーションデータが作成してあったからです。

モーションエディタは、いくつかのコマ(時間)の変数の値をユーザーが指定し(ツリーエディタで「ry　$0」のように変数指定をし)、その間のコマの変数の値が滑らかに変化するように自動的に補って、モーションデータを作成します。画面上では同時に3つまで、変数の時間による変化を表示できます。モーションエディタで指定したモーションデータの動作は、前述のようにポリゴンエディタでチェックできます。

カラーで表示

カラーで画像を表示してみましょう。「Mode」メニューの「画像表示」を選択します(または NFER ＋ G)。「recalc all animation (y/n)?」と聞いてきますので Y ⏎ を押すと、しばらくしてから画面にカラーの画像が表示されます(口絵⑲参照)。

Q を押すと MODE を終了して MS-DOS のコマンドラインにもどり、画像が画面に残ります。

アニメーション表示

続いてアニメーション表示させてみましょう。
まず次のコマンドを実行して、画面上に残っている画像を消します。

```
B>cls3 ⏎
```

そして、再び mode　sample ⏎ を実行して「sample.mod」を読み込んでください。

「Mode」メニューで「画像連続表示」を選択する(または NFER ＋ A)と、「recalc all animation (y/n)?」と聞いてきますので Y ⏎ を押します。しばらくしてから画面にカラーの画像が1枚1枚ゆっくりと表示されます(口絵⑳参照)。

「画像連続表示」も「画像表示」と同様に、 Q を押すと MODE を終了して MS-DOS のコマンドラインにもどります。

※フロッピーディスクに実行用ディスクを作成した場合は9コマ目までしか表示できません。

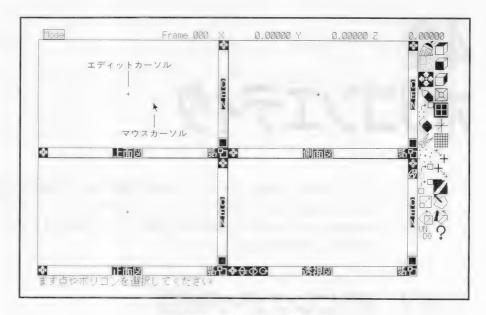
4章
ポリゴンエディタ

ポリゴンエディタではワイヤフレームを画面で確認しながら作成できます。このエディタによって作成されるデータは、モデルファイル中のポリゴンデータだけですが、さまざまな視点からこの3次元のデータを眺めて編集できます。配置するときはツリーエディタへ行き来しながら編集します。

4.1 ポリゴンエディタ画面

まずコマンドラインから次のように入力してMODEを起動しましょう。

B>mode ⏎

MODEを起動するとツリーエディタ画面が表示されるので、「Mode」メニューをドラッグして「ポリゴンエディタ」に反転表示を合わせ、マウスボタンを放します。ポリゴンエディタを選択すると次の画面が表れます。

画面 4-1　ポリゴンエディタの画面

「上面図」「側面図」「正面図」「透視図」には、それぞれ現在編集している物体の上面図、側面図、正面図、そして透視図(任意の方向から物体を眺めた図)を表示します。また、編集には画面右にある2列のアイコンを使います。

カーソルには、マウスの動きに合わせて移動するマウスカーソルと、各ウィンドウの中央に表示されているエディットカーソル(＋印)があります。マウスカーソルは画面内の座標を指定し、実行する作業の選択などに使います。エディットカーソルは、物体を作成する座標系内の座標を指定し、ポリゴンの頂点の座標の指定などに使います。エディットカーソルの座標は、画面の最上行に表示されています。

画面の最下行には、処理内容に応じたメッセージが表示されます。

4.2 ポリゴンの作成と削除

ポリゴンを作成するには、点を作り、点をつないでポリゴンを作る、という手順が必要です。まず、ポリゴンを作るために最低限必要なアイコンだけを使って正方形を作ってみましょう。使用するアイコンは　(選択点解除)、　(選択点指定モード)、　(カーソル移動モード)、　(点作成)、　(ポリゴン作成)、　(削除)、　(アンドゥ)です。

ポリゴンの作成

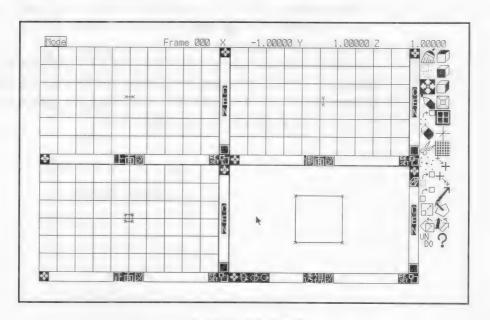
■点を作成

座標が整数値になるように（メッシュ）のアイコンをクリックしてメッシュを表示させます。

次にマウスカーソルを「正面図」の中に動かし、ドラッグして座標値が「1，1，0」を指したら、ボタンを放してください。そうするとエディットカーソルがマウスカーソルの位置に移動します。続いてマウスカーソルを「側面図」の中に動かし、ドラッグして座標値が「1，1，1」を指したら、ボタンを放してください。「側面図」のタイトルをクリックすると、エディットカーソルの位置に点ができます。同様にして「1，−1，1」、「−1，−1，1」、「−1，1，1」の座標で点を打ちます。

もし途中で違うところへ点を打ってしまったときは、UN DO（アンドゥ）のアイコンをクリックします。このアイコンは、直前の処理を取り消しでき、点が削除されます。

■選択点からポリゴンを作成

4点を指定したら、（ポリゴン作成）のアイコンをクリックします。各点が線で結ばれ、正方形のポリゴンができます。

画面 4-2　正方形の表示

ポリゴンの削除

■点を選択

いま点は赤い×印で表示されています。この点の状態を選択されているといい、この点を選択点といいます。ここで（選択点解除）のアイコンをクリックしましょう。そうすると、点が見えなくなります。この点の状態を選択されていないと呼びます。（選択点解除）のアイコンは、すべての点を選択されていない状態にします。

再びポリゴン上の選択されていない点を選択された状態にするには（選択点指定モード）のアイコンを使用します。（選択点指定モード）のアイコンをクリックしましょう。すると、（カーソル移動モード）のアイコンが通常表示になり、（選択点指定モード）のアイコンが反転表示になります。これは、エディットカーソルを動かせなくなる代わりに、点を選択できる状態になったことを示します。マウスカーソルを選択されていない点の上に持っていき、クリックしてください。点が作ったときと同じように赤い×印で表示されます。選択点をもう一度クリックすると、選択されていない状態になります。

■選択点の削除

今度は正方形を削除してみます。まず（選択点解除）をクリックして選択点をなくし、（選択点指定モード）のアイコンを反転表示させて消したい点を選択します。

すべての点を選択したら（削除）のアイコンをクリックしてください。点が削除されることで、ポリゴンも削除されます。

点を選択する方法には、点の上でクリックするほかにもう１つ方法があります。適当な位置からドラッグしてマウスを移動させると枠線が表示されます。枠線が選択したい点を囲む位置にきたらマウスボタンを放します。すると枠線が消え、枠線内の点が選択されます。こうすれば、ある範囲内の点すべてを選択できます。

4.3 ポリゴンの編集

この節では、前述の機能も含め、ポリゴンエディタの各機能についてアイコンを５つの種類に分けて説明をします。

- モードを選択するアイコン。ポリゴンエディタの２つのモードを切り替えます
- 選択点に対して処理を行うアイコン。ここに属するアイコンでデータを作成します
- 画面制御用のアイコン。これらには画面の表示状態を変える機能があります
- ウィンドウ上のアイコン
- その他のアイコン

次のアイコンはスイッチになっています。一度クリックするとアイコンが反転表示になり、そのアイコンの機能がオンになります。もう一度クリックすると、オフになり通常の表示にもどります。

（選択点指定モード）

（カーソル移動モード）

（透視図）

（正面図）

（側面図）

（上面図）

（座標軸）

（メッシュ）

（法線表示／非表示）

（HELP）

動作モード

ポリゴンエディタには2つのモードがあり、常にそのうちの1つが選択されています。

■選択点指定モード（　）

このアイコンをクリックすると、選択点を指定するモードになります。選択点は、選択したい点の上でクリックするか、マウスボタンを押したままマウスを動かして範囲を指定します。

点を作成するとき以外はこのモードにしておいてください。

■カーソル移動モード（　）

ポリゴンエディタで最初に選択されているアイコンです。このアイコンをクリックすると、エディットカーソルが移動できるモードになります。エディットカーソルの移動には2通りあります。ウィンドウ内で移動させたい座標でクリックして移動させるか、エディットカーソルをドラッグして細かい移動をさせるという方法です。（点作成）、「上面図」、「側面図」、「正面図」、「透視図」をクリックするとカーソル位置に点が打たれます。

選択点に対する処理

■選択点解除（　）

このアイコンをクリックすると、現在選択されている点すべてを選択されていない状態にします。同時に画面の描き直しも行います。

■ポリゴン作成（　）

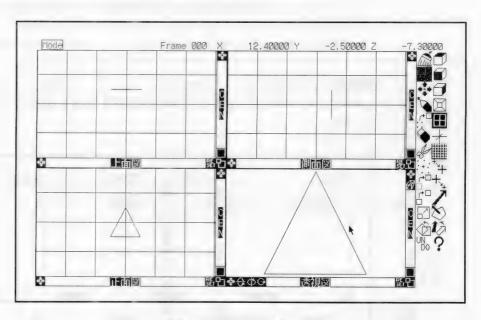
このアイコンをクリックすると、すべての選択点をつないでポリゴンを作成します。

■削除（　）

このアイコンをクリックすると、すべての選択点を削除します。

■中点作成（　）

このアイコンをクリックすると、2つの選択点の中間点に点を作ります。ただし、この2つの選択点はそれぞれ同じポリゴンの頂点となっていなければなりません。選択点が2つ以上ある場合も、この条件に合っていればおのおのの中間点に点を作ります。

この機能は、ポリゴンの頂点の数を変えたいときなどに使用します。たとえば、このアイコンで中間点を作り、その点を適当な位置に移動させます。

画面 4-3　正方形を三角形に変形

■移動（ ）

選択点指定モードの場合、選択点を移動させます。操作方法は、このアイコンをクリックし、移動させたい距離をエディットカーソルで指定し、もう一度このアイコンをクリックします。

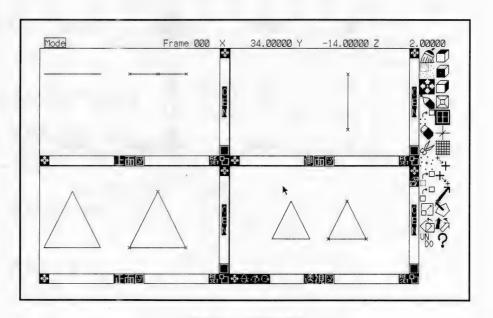
移動距離は、移動をはじめる前のエディットカーソルの座標から移動した距離となります。この距離は、通常エディットカーソルの座標が表示されている場所に表示されます。エディットカーソルの移動の方法は、カーソル移動モードでのエディットカーソルの移動の方法と同じです。

移動を中止したいときは、終了する前に UNDO（**アンドゥ**）をクリックします。

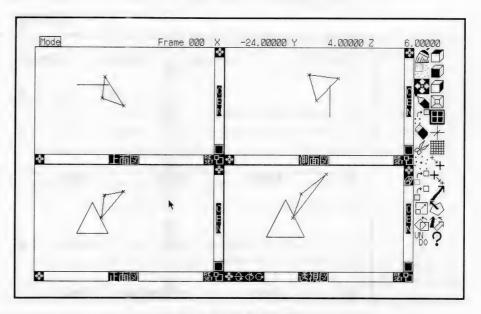
■複写（ ）

選択点、または選択ブロックを平行移動させて複写します。操作方法は （**移動**）と同じです。

画面 4-4　三角形を複写

■回転（ ）

選択点、または選択ブロックをエディットカーソルの座標を中心にして回転させます。操作方法は、このアイコンをクリックして回転角を指定し、もう一度このアイコンをクリックします。

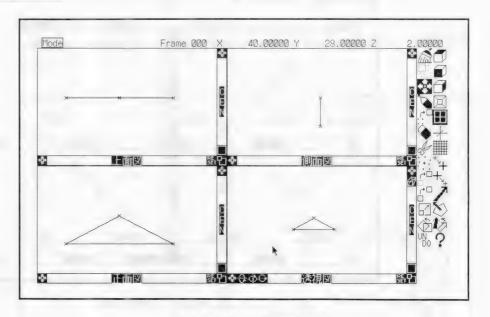
回転角は、エディットカーソルを移動させた座標で指定します。エディットカーソルを上面図で移動させた場合は、正面図に垂直で移動前のエディットカーソルの座標を通る直線を中心とした回転角を指定したことになります。

回転を中止したいときは、終了する前に UNDO（**アンドゥ**）をクリックします。

画面 4-5　三角形を回転

■拡大／縮小（　）

選択点、または選択ブロックをエディットカーソルの座標を中心にして拡大または縮小します。操作方法は、まずこのアイコンをクリックして拡大率(縮小率)をエディットカーソルで指定し、もう一度このアイコンをクリックします。

拡大率(縮小率)は、エディットカーソルを移動させた座標で指定します。移動前のエディットカーソルの座標をX、Y、Z座標につき1倍とし、各軸について正の方向に移動させるほど拡大され、負の方向に移動させるほど縮小されます。拡大率(縮小率)は、通常エディットカーソルの座標が表示されている場所に表示されます。

拡大または縮小を中止したいときは、UNDO(アンドゥ)をクリックします。

画面4-6　拡大で三角形を変形

画面制御用のアイコン

■三面図と透視図（ ）

三面図と透視図を表示します。ポリゴンエディタで最初に選択されている画面です。

■上面図（ ）、正面図（ ）、側面図（ ）、透視図（ ）

いずれかの図を画面いっぱいに表示します。

■座標軸（ ）

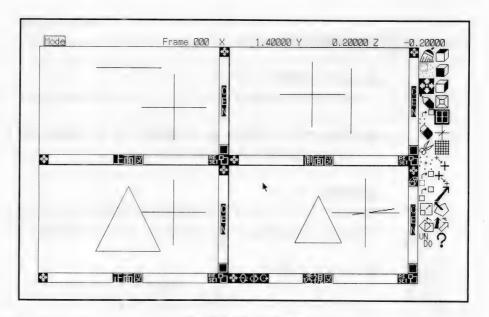
このアイコンをクリックすると、三面図および透視図上に原点を中心とした座標軸が表示されます。もう一度このアイコンをクリックすると、座標軸は消えます。

■メッシュ（ ）

このアイコンをクリックすると、三面図にメッシュを表示します。この状態では、エディットカーソルはメッシュ上以外には移動できません。このメッシュは、エディットカーソルが移動するすべての処理に適用されます。つまり、移動や拡大などの処理をするときにもエディットカーソルの動ける座標がメッシュ上のみとなります。メッシュを消すにはもう一度このアイコンをクリックします。

また、メッシュの間隔はエディットカーソルの大きさに依存します。

■カーソル拡大（ ）／カーソル縮小（ ）

エディットカーソルの大きさを拡大、縮小します。このアイコンを1回クリックすると1段階大きさが変化します。表示のスケールを変えたためカーソルが適当な大きさで表示されなくなったときに使用します。また、メッシュの間隔を変えたいときにも使用します。

画面 4-7　カーソルを拡大

ウィンドウ上のアイコン

■点作成（「上面図」「側面図」「正面図」「透視図」）

各ウィンドウの中の「上面図」や「側面図」、「正面図」、「透視図」をクリックすると、エディットカーソルの座標に点を作成できます。作られた点は、選択された状態になっています。

■上下左右移動（ ）

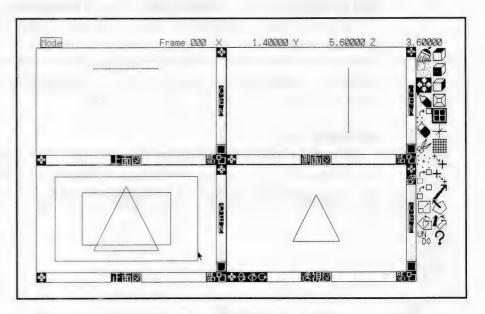
大きな物体を作ったときなどは、ウィンドウの中に物体の全体が入りきらないときがあります。そういうときは、このアイコンを使って表示される座標の範囲を指定します。

このアイコン上でドラッグすると、ウィンドウ内に四角い枠が表示されます。この枠がマウスの動きに合わせて動くので、表示させたい範囲へ移動させてマウスボタンを放します。表示範囲は枠を動かした方向と逆に移動します。

■リセット（RESET）

このアイコンをクリックすると、図形を初期状態にもどします。たとえば、表示範囲の中心を原点にする、スケールをもとにもどす、視点を原点にして方向はZ軸方向にする、などの処理を行います。

■スケール変更（　）

表示する物体の縮尺を変更します。操作方法は、このアイコン上でドラッグすると画面に長方形の枠線が表示されます。このままマウスを動かして枠線の大きさを変化させて縮尺を決定し、ボタンを放します。この枠線の大きさを小さくすれば拡大、大きくすれば縮小します。

画面 4-8　スケール変更の枠線

■フィット（ □ ）

　このアイコンをクリックすると、縮尺を変更して編集中のすべてのポリゴンがウィンドウの画面いっぱいに表示されます。三面図のいずれかで行ったときは、三面図すべてが画面いっぱいに表示されます。「透視図」で行ったときは、「透視図」だけが画面いっぱいに表示されます。

■センター（「CEN」）

　各ウィンドウ上の「CEN」(**センター**)アイコンをクリックすると、エディットカーソルを中心として表示します。

■X軸視点回転（ ）、Y軸視点回転（ ）、Z軸視点回転（ ）

　このアイコンを使って、透視図の視点をエディットカーソルからX軸(Y軸、Z軸)に平行に伸ばした線を中心にして回転させます。このアイコン上でドラッグしてマウスを左右に動かしてマウスボタンを放すと、透視図上を線が回転します。この線の向きで回転角を決定します。この機能を使う前に、エディットカーソルを編集中の物体の中心に移動させておくとよいでしょう。

■前後移動（ ）

　透視図の視点を移動します。操作方法は、このアイコン上でドラッグすると画面に長方形の枠が表示されるので、このままマウスを動かして長方形の大きさを変更し、ボタンを放します。この長方形の大きさを小さくすれば視点を前方へ移動、大きくすれば後方へ移動します。

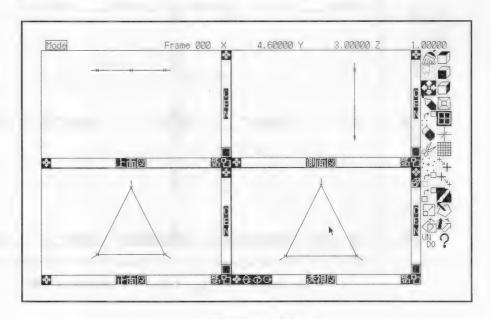
その他のアイコン

■法線生成（ ）

法線ベクトルを生成します。法線ベクトルは、画像ファイルを作るときにポリゴンの頂点を滑らかに見せるために指定します。

法線ベクトルを表示するためには、まずこのアイコンを選択して法線ベクトルを生成する必要があります。

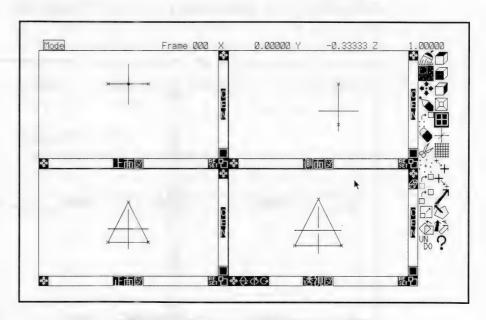
■法線表示（ ）

法線ベクトルの表示、非表示の切り替えを行います。（法線生成）をクリックして法線生成した後、続けてこのアイコンをクリックすると法線を表示します。

画面 4-9　法線を表示

■法線変更（　）

三面図を使って法線ベクトルの向きを変更します。変更するときは、まず点を選択(赤色の×印を表示)してから　(法線表示)をクリックして反転表示させ、このアイコンをクリックします。選択点から赤色の線が表示されるので、ドラッグしてエディットカーソルを動かしながら法線ベクトルの向きを変更します。もう一度　(法線変更)をクリックすると法線ベクトルの向きが決定して緑色の線で表示されます。

■センタリング（　）

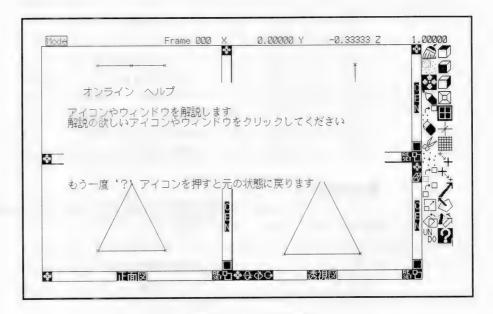
このアイコンをクリックすると、エディットカーソルをすべての選択点の中心へ移動させます。

画面 4-10　エディットカーソルを三角形の中心に移動

■アンドゥ（UNDO）

直前の処理を取り消します。2度続けてクリックすると、1度目のアンドゥを取り消します。

■HELP（?）

このアイコンをクリックした後に他のアイコンをクリックすると、そのアイコンの機能の簡単な説明が表示されます。HELPを終了するには、もう一度このアイコンをクリックします。

また ?(HELP)が反転表示中に他のメニューを選択した場合にもHELPを終了します。

画面4-11　HELPの表示

4.4 キーボードによる操作

ポリゴンエディタでできるキー操作は「Mode」メニューの選択、動画表示の2つです。

「Mode」メニューの選択については**ポリゴンエディタ、ツリーエディタ、モーションエディタ、ファイルの切り替え**で共通です。

■「Mode」メニューの選択

ポリゴンエディタではマウスによる「Mode」メニューの選択を次のキー操作で代用できます。

キー	機能
NFER ＋ T （XFER ＋ T）	ツリーエディタへ
NFER ＋ P （XFER ＋ P）	ポリゴンエディタへ
NFER ＋ M （XFER ＋ M）	モーションエディタへ
NFER ＋ F （XFER ＋ F）	ファイルの切り替え
NFER ＋ S （XFER ＋ S）	シェルの起動
NFER ＋ G （XFER ＋ G）	画像表示
NFER ＋ A （XFER ＋ A）	画像連続表示
NFER ＋ Q （XFER ＋ Q）	MODEの終了

■動画表示

ポリゴンエディタにはモーションエディタで作ったモーションデータの動きを確認する機能もあります。次のキーでコマの確認をします。

キー	機能
N	次のコマを表示
スペース	次のコマを表示
P	前のコマを表示
CTRL ＋ H	前のコマを表示
CTRL ＋ L	画面を再描画
SHIFT ＋ ?	キーの説明を表示

なお、モーションデータがないときに スペース を押すと、画面を再描画します。

5章
ツリーエディタ

ツリーエディタはMODEの中心となるものです。ポリゴンエディタで作ったポリゴンの配置を決めたり、モーションエディタで作ったモーションデータで動かす部分を指定したりします。おもな機能は次のとおりです。

- ポリゴンの表示位置指定
- 他のポリゴンデータの読み込み
- ブロックの作成
- ツールの使用
- モーション部分の指定
- ポリゴンの色指定(第3編参照)

5.1 ツリーエディタ画面

MODEは、常にツリーエディタから起動します。「3章　MODEの概要」で紹介した「sample.mod」の画面で簡単に解説してみましょう。

画面5-1　ツリーエディタ画面

```
sample.mod─┬─x 213,64,0,1;       ……………… 横方向の解像度を213ドット、視界を64度に指定
           ├─y 133,40,0,1;       ……………… 縦方向の解像度を133ドット、視界を40度に指定
           ├─l 0,0.5,0.5,0.5     ……………… 背景色を指定(灰色)
           ├─l 1,1,1,1       ……………… 光源色を指定(白色)
           ├─b 0,(0,0,0)     ……………… 光源の向きを指定
           ├─b 1,(0,5,-1)……………… 光源を指定
           ├─tz 30       ……………… 物体をZ軸にそって30移動
           ├─ry $0:0     ……………… 物体をY軸を中心にして$0(変数)度回転
           ├──┬─# 球     ……………… コメント
           │  ├─c 36     ……………… 球を赤色に指定
           │  ├─ty 5     ……………… 球をY軸にそって5移動
           │  └─sx 2     ……………… 球の半径をX方向に2倍にする
```

画面にはモデルデータが文字で表示されます。この1行が1つのモデルデータであり、同じブロックに属するデータが線で結ばれています。ブロックにされたデータは、そのブロックのレベルが低いほど右に表示されます。またモデルデータの最上行の左には、モデルデータのファイル名が表示されます。

カーソル(■)がモデルデータの左に表示されていますが、このカーソルで処理の対象となるデータを指定します。また、このカーソルの位置によって、ポリゴンエディタで表示されるデータの範囲が変わります。表示されるのは、カーソルが指定しているデータの属するブロックだけです。

5.2 モデリングの実際

この節ではツリーエディタを使ってポリゴンエディタで確認しながらモデリングしてみます。

基本的な操作

ツリーエディタではおもにキーボード入力で操作を行いますので、まず基本的なキー操作を以下に示します。ただし最低限必要なキーは[H]、[L]、[J]、[K]の4つだけなので他のキーは必要に応じて覚えていけばよいでしょう。詳しいキー操作は「5.3　データの編集」で説明します。

なお、アルファベットは大文字([SHIFT]＋と記述)と小文字を区別するので気を付けてください。

カーソル移動キー

キー操作	内　容
[J]	次(下)のデータへ移動
[K]	前(上)のデータへ移動
[H]	1つ上のレベル(左)へ移動
[L]	1つ下のレベル(右)へ移動

データ編集キー

キー操作	内　容
[SHIFT]+[>]	カーソル位置のデータのレベルを１つ下げる(ブロック操作に使用)
[SHIFT]+[<]	カーソル位置のデータのレベルを１つ上げる(ブロック操作に使用)
[D]	カーソル位置のデータを削除し、バッファへためる(ブロック操作に使用)
[P]	バッファのデータをカーソル位置の次の行へ挿入(ブロック操作に使用)
[O]	ダミーのデータををカーソル位置の次の行へ挿入(データの書き込みに使用)
[SHIFT]+[C]	データのタイプを変更(データの書き込みに使用)
[C]	データの引数を変更(データの書き込みに使用)
[SHIFT]+[$]	データの引数が定数か変数かを選択
[U]	直前の処理を取り消す

キー操作が分からなくなったときは、[SHIFT]＋[?]でツリーエディタのヘルプ画面を表示して確認しましょう。

```
使用方法: このモードでのキー割当て
 ^B: 前のページへ移動する
 ^D: 画面半分次のページへ移動する
 ^F: 次のページへ移動する
 ^L: 画面を再描画する
 ^R: カーソルが画面の中心にくるように再描画する
 ^U: 画面半分前のページへ移動する
' ': カーソルがiコマンドの上にあればその対象であるファイルを編集する
     カーソルがpコマンドの上にあればその内容を表示する
  C: カーソルのある枝を変更する
  c: カーソルが示す値を変更する
  d: 枝を削除する
  e: 他のファイルを編集する
  h: カーソルを左に動かす
  J: 枝があればカーソルを下に動かす
  j: カーソルを下に動かす
  K: 枝があればカーソルを上に動かす
  k: カーソルを上に動かす
  l: カーソルを右に動かす
  m: マークする
  ': マークへ移動する
  O: 新しい枝をカーソルの上に挿入する
  o: 新しい枝をカーソルの下に追加する

何かキーを押してください
```

画面 5-2　ツリーエディタのヘルプ画面

■"i"コマンドで床の読み込み

まず mode ⏎ と入力して MODE を起動すると次の画面が表示されます。

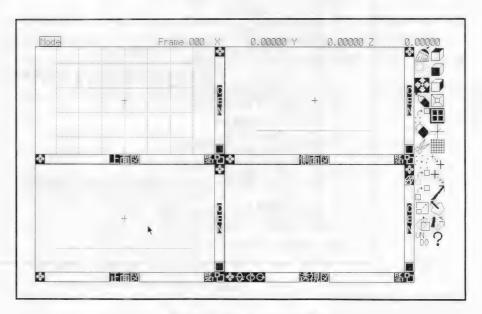
画面 5-3　ツリーエディタ画面

```
█── NONE
```

C（データの引数を変更）、I（"i"コマンドの入力）、L（1つ下のレベルへ移動）、C と続けて floor.mod ⏎ H と入力します。そうすると次の画面が表示されます。

画面 5-4　ツリーエディタ画面

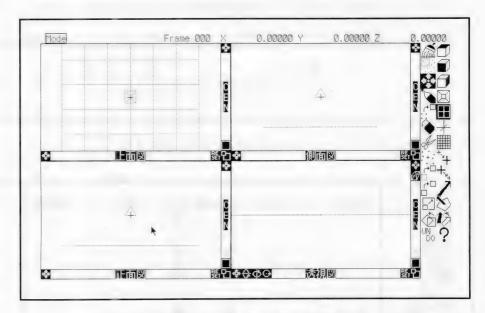
```
█── i floor.mod
```

ここで「ポリゴンエディタ」を選択すると床のワイヤフレームが表示されます。

画面 5-5　ポリゴンエディタ画面

■床の上に四角錘を配置

今度は床の上に四角錘を配置してみましょう。

「ツリーエディタ」を選択して、SHIFT＋O で「NONE」を作成します。先程と同様に C、I、L、C とキー操作して pyramid.mod ↵ と入力します。

そして H キーでカーソルを先頭位置に移動して、「ポリゴンエディタ」を選択して画面を確認します。

画面 5-6　ポリゴンエディタ画面

■"ｔ"コマンドで位置を移動

画面では床と四角錘が離れて表示されていますので、くっつけてみます。

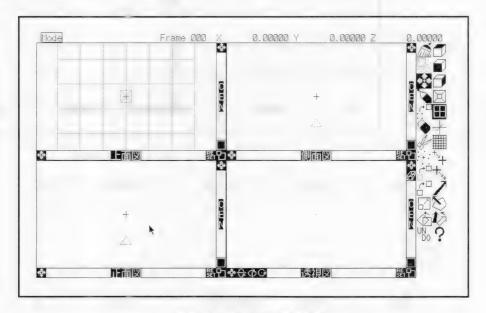
（メッシュ）を選択して「正面図」でマウスカーソルを床の上に移動します。Y座標が「−5」なので、Y方向に「−5」移動すればよいことが分かります。

「ツリーエディタ」を選択し、最初に SHIFT ＋ O で「NONE」を作成します。C 、T 、Y 、L 、C とキー操作して−5 ⏎ と入力します。

カーソルは H キーで先頭位置にもどしておきましょう。

画面 5-7　ツリーエディタ画面

```
█── ty -5
 ├── i pyramid.mod
 └── i floor.mod
```

「ポリゴンエディタ」を選択して画面を確認します。

画面 5-8　ポリゴンエディタ画面

四角錘だけを移動したつもりが、床にくっついていません。これは"t"コマンドが両方にきいてしまったため、全体がY方向に「−5」移動してしまったのです。

■"＞"コマンドで階層化

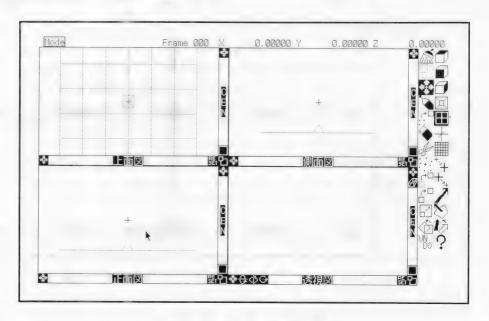
それでは四角錘だけを移動するときはどうすればよいでしょうか。一部にだけそのコマンドを有効にさせるためには、その部分をブロックに(階層化)します。

「ツリーエディタ」を選択して [SHIFT] ＋ [>] とすると、「ty　－5」が1つ下の階層に移動します。次に [J] キーでカーソルを移動して [D]、[K]、[P] と続けて入力して四角錘のデータを「ty　－5」と同じブロックに移動します。

終わったら [H] キーでカーソルを先頭位置にもどしておくのは前述のとおりです。

画面 5-9　ツリーエディタ画面

```
█────ty -5
 │  └──i pyramid.mod
 └──i floor.mod
```

「ポリゴンエディタ」を選択すると、今度は四角錘が床にくっついて表示されます。

画面 5-10　ポリゴンエディタ画面

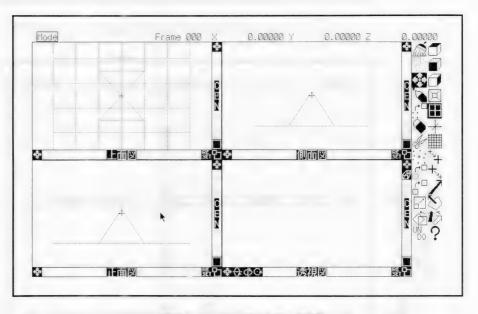
■"s"コマンドで四角錘を拡大

四角錘が床に対して小さすぎるので、拡大してみましょう。拡大には"s"コマンドを使います。

「ツリーエディタ」で、[L]キーでカーソルを「ty　−5」の左に移動して[O]キーで「NONE」を作成します。[C]、[S]、[X]、[L]、[C]とキー操作して4[↵]と入力します。これでX軸にそって4倍に拡大します。

同様に[O]、[C]、[S]、[Y]、[L]、[C]とキー操作して4[↵]と入力すればY軸にそって4倍に拡大します。

また、[O]、[C]、[S]、[Z]、[L]、[C]とキー操作して4[↵]と入力すればZ軸にそって4倍に拡大します。

[H]キーを2回押して先頭位置にカーソルをもどして「ポリゴンエディタ」を選択すると、四角錘が床の上に拡大して表示されます。

画面5-11　ポリゴンエディタ画面

ファイルを保存

これまで作ってきたファイルを「test.mod」というファイル名で保存しておきましょう。

「Mode」メニューで「終了」を選択すると、「"というファイルをセーブしますか？(y　or n)」と聞いてきますので、Y キーを押します。

続いて「セーブするモードのファイル名は：」と聞いてくるので test.mod ⏎ と入力します。

5.3　データの編集

この節では、前述の機能も含め、ツリーエディタの各機能について解説します。

キーの種類

これまで説明してきたように、ツリーエディタではキーボードの操作で編集をします。入力は、アルファベット大文字、小文字、記号、コントロールキーとアルファベットキーの併用などの方法があり、それぞれのキーに機能が割り当てられています。

ブロックの操作は、SHIFT ＋ > や SHIFT ＋ < のキーを使います。また、データを削除したり作成したりするために、D 、SHIFT ＋ O 、O 、SHIFT ＋ P 、P 、Y などの編集キーを使用します。

データの書き込みには、SHIFT ＋ O 、O 、SHIFT ＋ P 、P 、Y 、SHIFT ＋ C 、C などのキーを使います。

ポリゴンエディタで表示するデータは、ツリーエディタのカーソル位置によって変わります。表示されるのは、カーソルが位置するブロック内で、カーソルより下に表示されるデータです。

他のデータファイルを編集するときは、E キーで読み込みます。また、現在編集中のファイルの中から編集するファイルを指定するには、カーソルをそのファイル名に合わせて スペース キーを押します。

ポリゴンはおもにポリゴンエディタ上で作成しますが、たとえば、球のデータをポリゴンエディタで作るのは非常に大変です。こういった処理は専用のツールを使えば簡単に作成できます。ツールを使用するときは、SHIFT ＋ ! キーを押してコマンド入力モードにし、画面最下行の「実行するコマンド：」に続けてコマンドを入力します。ツリーエディタで使用できるツールについては「Appendix D　コマンドリファレンス――ツリーエディタで使えるツール」を参照してください。

■カーソル移動キー

カーソルの移動方法には、以下のようなものがあります。

キー操作	内　容
CTRL + B	1画面上に移動
CTRL + U	半画面上に移動
CTRL + F	1画面下に移動
CTRL + D	半画面下に移動
CTRL + R	カーソルが画面の中心になるように表示
H	1つ高いレベル(左)へ移動。最上位レベルで押すと、「ファイルの切り替え」へ移動
L	1つ低いレベル(右)へ移動
J	1つ下へ移動
K	1つ上へ移動
SHIFT + J	1つ下のブロックへ移動
SHIFT + K	1つ上のブロックへ移動
M	カーソル位置に印を付ける
'	印を付けたデータに移動

■編集機能キー

O

いまカーソルが指しているデータの後ろに「NONE」というダミーのデータを挿入します。この機能は、データを書き込むときに使用します。データを書き込むときは、まずこのキーで「NONE」を作り、それから SHIFT ＋ C キーで他のデータに変更するという操作をします。

SHIFT ＋ O

いまカーソルが指しているデータの前にダミーのデータを挿入します。O との違いは、データのできる位置が異なるだけです。

P

バッファにあるデータを取り出し、カーソル位置の後ろに挿入します。取り出した後もバッファ内にデータが残っているので、同じデータを好きなだけ挿入できます。

SHIFT + P

バッファにあるデータを取り出し、カーソル位置の前に挿入します。P との違いは、データの挿入される位置が異なるだけです。

D

カーソル位置のデータを削除し、バッファに入れます。実行すると、この機能を実行する前にバッファにあったデータはなくなります。

Y

カーソル位置のデータを削除せずに、バッファに入れます。実行すると、この機能を実行する前にバッファにあったデータはなくなります。

SHIFT + >

カーソル位置のデータのレベルを1つ下げ、新しいブロックを作ります。

SHIFT + <

カーソル位置のデータのレベルを1つ上げ、カーソル位置のブロックを消します。ただし、実行できるのはカーソル位置のブロックにデータが1つあるときだけです。

C

データの引数を変更します。引数をとるデータは、ポリゴン以外のすべてのデータです。適当な引数を入力した後に改行を押すと、引数の変更が終了します。また、引数の変更を中止するには、ESC キーを押します。

C キーを押すと引数が数値をとるコマンドの場合は変更前の引数は表示されませんが、引数が文字列をとるコマンドの場合は変更前の引数が表示されます。文字列を修正したい場合は、CTRL + F や CTRL + B でカーソルを移動させ、間違えた文字を CTRL + D で削除し、正しい文字列を入力します。文字列をすべて入力し直したい場合は、CTRL + K で引数をすべて削除した後に新しい引数を入力します。

次に引数入力に使用できる文字とキーを示します。

使用できる文字	説　明
012−9	数値やファイル名に使用
.	小数点を表す
;	不定個数の引数列を要求するコマンドで、終了を表す
−	負の数値を表す

使用できるキー	説明
↵	引数を確定して終了
ESC	引数の入力を中止して終了
CTRL+F	カーソルを右へ移動
CTRL+B	カーソルを左へ移動
CTRL+H	カーソルの前の1文字を削除
CTRL+D	カーソル位置の1文字を削除
CTRL+U	カーソルの前の文字をすべて削除
CTRL+K	カーソルの後ろの文字をすべて削除

SHIFT＋C

データのタイプを変更します。この機能で変更できるデータは、ポリゴン以外のすべてのデータです。この機能を実行すると、画面左下に入力できるデータの種類が表示されるので、入力したいデータを選んで入力します。データによってはさらにもう一度入力するものもあります(r、t、sコマンド)。入力が終わるとデータのダミーの引数が設定されるので、cコマンドで希望する数値に変更します。

SHIFT＋$

データの引数には定数か変数を使用できるので、この機能で選択します。いまの引数が定数のときは変数に、変数のときは定数に変更します。引数が定数のときはその値が表示され、変数のときは$の後に0から9までの数字のいずれかが表示されます。変数は0から9までの10個が使用でき、cコマンドで選択します。変数の値は、モーションエディタで作成されたパラメータファイルによって与えられます。

■その他のキー操作

キー操作	内　容
スペース	指定したデータがポリゴンデータならばその座標データを表示。「i　〈ファイル名〉」ならば「ファイル名」のファイル編集画面に移動
CTRL + L	画面を再描画
U	1つ前の動作を取り消す
E	モデルファイルのファイル名を入力し、他のモデルファイルを読み込む
W	データを画面上のファイル名で保存
R	ファイル名の変更
SHIFT + !	ツールを呼び出す
SHIFT + ?	操作方法の一覧を表示

応用操作

■繰り返し処理

繰り返し同じ操作を実行させるときは、操作の前に数字キーで実行回数を入力します。このとき入力した数字は表示されません。

たとえば、10d と入力すれば、カーソル位置から下10行分のデータが削除されますが、10行データがなかった場合は、ある分だけ削除されます。この性質を利用して、カーソル位置から下をすべて削除したい場合は、100d など実際の行数より大きい数字を入力します。

ただし、ブロックの中で実行したときにはブロック内だけで削除されます。

■バッファ

データの移動や複写には、バッファを利用します。バッファとは一時的にデータをためておく場所のことで、一度バッファに入ったデータは新たにデータをバッファに入れるまで消えません。

データを移動させるには、移動させたいデータを D キーで削除します。このとき削除したデータはバッファにためられています。次にカーソルを移動先に置き、P（または SHIFT + P）キーを押すとデータ移動が完了します。

データを複写するには、D キーの代わりに Y キーを使用します。続けて移動の場合と同様に複写先で複写したい数だけ P（または SHIFT + P）キーを使用します。

5.4　コマンドの種類

ツリーエディタでは、ポリゴンデータの表示位置や大きさを変更できます。ここでは入力するコマンドデータの種類とその意味および引数の書式を説明します。

コマンドデータの引数には整数か実数または文字列をとりますが、引数が整数か実数の場合は定数を使用する代わりに変数を使用できます。変数の実際の値は、モーションエディタで出力されるパラメータファイルで与えられます。

コマンドには次のような種類があります。

コマンド	内　容
#	コメント
i	ファイルの読み込み
r	回転移動コマンド
t	平行移動コマンド
s	拡大、縮小コマンド
b	ブリンシェーディング(第3編参照)
c	表面モデル用コマンド(第3編参照)
l	光源モデル用コマンド(第3編参照)
x	横方向の解像度(第3編参照)
y	縦方向の解像度(第3編参照)
S	色拡張コマンド(このパッケージでは使いません)

ポリゴンの配置

■平行移動コマンド (Translation)

書　式	t[xyz]　〈距離〉
例	tx　30

tコマンドにはtx、ty、tzの3つがあり、それぞれX、Y、Z方向にデータを平行移動させます。距離は実数で与えます。

■回転移動コマンド（Rotation）

書　式	r[xyz]　〈角度〉
例	ry　80

rコマンドにはrx、ry、rzの3つがあり、それぞれX軸、Y軸、Z軸を中心にしてデータを回転させます。角度の単位は度数を実数で与え、360度で1回転です。

■拡大、縮小コマンド（Scaling）

書　式	s[xyz]　〈倍率〉
例	sz　60

sコマンドにはsx、sy、szの3つがあり、それぞれX軸、Y軸、Z軸方向に原点を中心として与えた倍率だけ拡大されます。倍率には0以外の実数を与えます。倍率に1未満の値を与えれば縮小し、負の値を与えれば各軸をはさんだ反対側に倍率の絶対値だけ拡大します。

■コマンドの順番

t、r、sといった座標変換コマンドは、コマンド以降に記述されたデータに対して影響を及ぼしますが、データに近い順に作用することに注意してください。たとえば後述するファイル読み込みコマンド"i"を使うと中心(1，0，0)、半径0.5の球は次のようになります。

```
tx  1              X方向に1移動
sx  0.5            X軸を0.5倍
sy  0.5            Y軸を0.5倍
sz  0.5            Z軸を0.5倍
i  sphere.mod      中心(0，0，0)、半径1の球のデータ
```

これを次のようにtxコマンドの位置を変えて指定すると、X方向に1移動してから縮小するので中心が(0.5，0，0)となってしまいます。

```
sx  0.5
sy  0.5
sz  0.5
tx  1
i  sphere.mod
```

ファイル読み込み用コマンド（Include）

書　式	i 〈ファイル名〉
例	i　model.mod

〝i〟コマンドは、他のモデルファイルを読み込むコマンドです。また〝i〟コマンドで読み込んだファイルから、さらに〝i〟コマンドで他のファイルを読み込むこともできます。

〝i〟コマンドを使うとき、正確なファイル名が思い出せない場合があります。そういうときに、「シェルの起動」を使えばMODEを終了せずにファイル名を確認できます。

「Mode」メニューで「シェルの起動」を選択すると、いったんMS-DOSのコマンドラインにもどります。

たとえばdir　*.mod⏎のように入力してファイル名を確認し、MODEにもどるときはexit⏎と入力します。

MODEにもどるとシェルを起動する前の画面を表示します。

コメント

モデルファイル中にコメントを入れるためには、〝#〟を使います。コメントはモデルファイル内のどこにでも置くことができ、〝#〟の後に他のコマンドが指定されていても〝#〟から行末までは無視され、実際の処理には影響しません。

画面5-12　コメント

```
├─#解像度と視野角度の指定
├─x 80,40,0,1;
├─y 50,25,0,1;
├─#背景色の指定
└─l 0,0,0,0
```

階層化コマンド（>、<）

座標変換コマンドは一度指定するとデータの最後まで影響を及ぼしますが、階層化コマンドを使って効果を与える範囲を制限できます。それには〝>〟と〝<〟を使用します。

階層化コマンドで囲まれた範囲をブロックと呼び、ツリーエディタではブロックごとに段を付けて表示します。

ブロック内で指定されたコマンドは、ブロックの中のデータにのみ有効です。しかし、ブロックの外で指定された座標変換コマンドはブロックの中へも影響を及ぼします。

■ブロックの作成

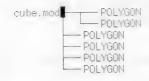
部品ごとに色や配置を変えられるようにブロックを作成します。

たとえば「cube.mod」を読み込むと、画面には6行、POLYGONと表示されます。

最初に、上から2つのデータを1つのブロックにしてみます。これには、1行目のデータで新しいブロックを作り、2行目のデータを新しいブロックに移動させるという手順が必要です。

K キーを押してカーソルをいちばん上の行に移動させます。次に SHIFT ＋ > キーを押して、カーソル位置のデータで新しいブロックを作ります。そうすると、カーソル位置のデータが右に移動して表示されます。ツリーエディタでは、ブロックによる階層化の深さ（レベル）を表示位置で表現します。右に表示されるデータほど深い階層のブロックに属します。

次に2行目のデータをいま作ったブロックに移動させます。J と入力してカーソルを2行目に移動させ、D キーを押すと2行目のデータは削除されます。次に、K と入力してカーソルを1行目のデータにもどし、P キーを押すと削除したデータがカーソルの次の行に表示されます。

画面 5-13　ブロックの作成①

D キーによって削除されたデータは一時的にバッファに保存されます。バッファとは、一時的にデータを保存しておく場所のことで、P キーによってバッファ内のデータをカーソルの次の行に挿入できます。これを利用してデータの移動を行ったわけです。

操作中に間違えてデータを消してしまったら、すぐに U キー（アンドゥ）を押してください。そうすると、ポリゴンエディタと同様に直前のコマンドが取り消されます。

今度は残りの4つのデータでブロックを作ります。同じ様に3行目にカーソルを移動させ、SHIFT ＋ > キーを押します。次に4行目にカーソルを移動させ、今度は 3 、D と入力すると、3行分のデータが一度に削除されます。続けて L 、P と入力すると3行分のデータがカーソルの次の行から表示されます。

画面 5-14　ブロックの作成②

このようにコマンドの前に数字を入力すると、指定回数だけコマンドを繰り返して実行できます。

■ブロックの削除

ブロック内のすべてのデータを1つ上のレベルのブロックに移動させるには、ブロックを作るときと逆の操作をします。まずブロック内のいちばん上のデータを残して削除し、残ったデータ（いちばん上にあったデータ）を SHIFT ＋ < キーで移動させ、移動させたデータの下に削除したデータを挿入します。

1行目と2行目のデータをもとの位置にもどしてみます。カーソルを2行目に移動させ、D キーを押して、カーソル位置のデータを削除します。次にカーソルを1行目に移動させ、SHIFT ＋ < キーを押してカーソル位置のデータを1つ上のレベルのブロックに移動させます。最後に P キーを押して削除したデータを挿入してください。

画面 5-15　ブロックの削除

5.5　モデルデータの挿入

新たにデータを書き込むには、まず書き込みたい位置にカーソルを合わせ、O キーを押します。そうするとカーソル位置の次の行に「NONE」というデータが書き込まれます(データの先頭に書き込む場合には、SHIFT ＋ O キーを使用します)。「NONE」とは、モデルデータが空のダミーのデータです。次に C キーでデータのタイプを変更します。ここでは「tx　10」というデータを書き込んでみます。C キーを押すと、画面左下に変更できるデータのタイプが表示されるので、これを見て t を入力します。次に「x, y, z」と表示されるので、x と入力すると、「NONE」は「tx　0」に変更されます。tx の引数はこの時点で自動的に 0 に設定されるので、これを 10 に変更しましょう。L キーを押してカーソルを引数の位置に合わせ、C キーを押して 10⏎ と入力します。ここで入力を間違えた場合は、BS キーを押すと修正できます。

画面 5-16　cube.mod を編集

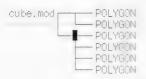
いま引数を定数で書きましたが、これを変数に変更してみましょう。カーソルを引数に合わせて SHIFT ＋ $ キーを押すと、引数が「$0」に変更されます。次に C キーで「$」の後ろの数字を任意の数字に変更し、使用する変数を選択します。変数は 0 から順に使用してください。変数を定数に変更するときは、カーソルを引数に合わせて SHIFT ＋ $ キーを押します。

5.6 カーソルの位置

カーソルの位置には3つの意味があります。ツリーエディタでコマンドの対象となるデータの指定、ポリゴンエディタで表示するブロックの指定、そしてポリゴンエディタでポリゴンを作る位置の指定です。

ツリーエディタでのデータ指定の方法は、これまでの操作で理解できたと思います。そこで、ツリーエディタのカーソルがポリゴンエディタでどのように影響するか実際に操作してみます。

次の例でカーソルを3行目のデータを指定する位置に移動させてポリゴンエディタを選択してください。そうすると、ポリゴンが4つだけ表示されます。

画面 5-17　カーソルの位置

ポリゴンエディタでは、ツリーエディタの画面でカーソルのあるブロックに属しているものだけを表示するのです。次にツリーエディタにもどってカーソルを1行下に移動させ、またポリゴンエディタを選択してください。今度は3つのポリゴンが表示されます。つまりポリゴンエディタでは、カーソルの位置するブロック内で、カーソルより下に表示されるデータを表示します。この性質を利用して編集中のブロックだけを表示させれば、複雑なデータを作っているときでも見やすい状態で編集ができます。

またポリゴンエディタで作ったポリゴンはツリーエディタでのカーソル位置の前に挿入されます。

5.7 ツール

ポリゴンを作成するには、ポリゴンエディタを使用するほかに、ツリーエディタでツールを使用する方法があります。ツールを使用すると、複雑な図形なども簡単に作れるようになります。

■ツールの種類

ツリーエディタから使用できるツールには、平面(PLANE)、球(SPHERE)、円柱(TUBE)、円錐(CONE)、多面体(POLHD)を作るコマンドなどがあります。

ツリーエディタから SHIFT ＋ ! キーを押すとツールの指定画面が表示されます。

```
ツールの使い方
---------------
コマンド名：  PLANE  選択点の順番でポリゴンを生成

書式          PLANE    [-]      [<ファイル名>]

機能
  PLANEコマンドは、選択点の順番でポリゴンを生成します。" -" を指定すると結果を
標準出力に出力し、<ファイル名>を指定すると結果をファイルに出力します。

オプション
  -         結果を標準出力に出力する。

解説
  ポリゴンエディタを使ってポリゴンを作ると、選択点を結ぶ順番が自動的に決められ
てしまうため、凹部のあるポリゴンを作るときなどは多少手間がかかります。PLANEコマ
ンドでは選択点を作った順にポリゴンが作成できます。<ファイル名>を指定するとき拡
張子（*.mod）は省略できます。

サンプル・オペレーション
  plane  sample
  plane  -

実行するコマンド: ■
```

画面 5-18　"!"コマンド画面(2ページ目)

ここで ⏎ キーを押していくと、ツリーエディタで使えるツールの説明が順次表示されます。「書式」のとおりにツールの名前とその引数を入力して ⏎ キーを押します。実行すると、現在編集中のデータにツールが出力するデータが付け加えられます。

ここではPLANEコマンドとSPHEREコマンドを例として解説します。その他のコマンドについては「Appendix D　コマンドリファレンス――ツリーエディタで使えるツール」を参照してください。

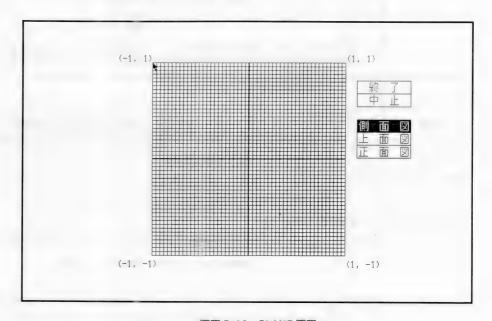
平面を作成（PLANE）

ポリゴンエディタを使ってポリゴンを作る場合、ポイントを結ぶ順番は自動的に決められてしまうため、凹部があるポリゴンを作ろうとすると多少手間がかかります。そういうときのためにplaneコマンドがあります。

■起動と終了

ツリーエディタから SHIFT ＋ ! キーを押すと画面最下行に「実行するコマンド：」が表示され、入力を促しますので、次のように入力してください。

plane　─⏎

そうすると次の画面が表示されます。

画面 5-19　PLANE 画面

また、MS-DOSのコマンドラインから次のようにして結果をファイルに出力できます。

plane　〈ファイル名〉⏎

終了するときは、画面右上の「終了」をクリックします。そうすると、ポリゴンデータを出力して終了します。また「中止」をクリックすると、データを出力せずに終了します。

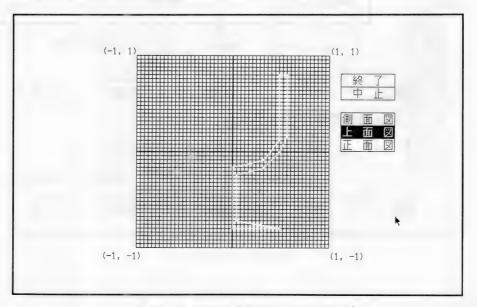
※ PLANE コマンドでは既存のファイルの編集ができません。すでに存在する〈ファイル名〉を指定すると上書きされますので、注意してください。

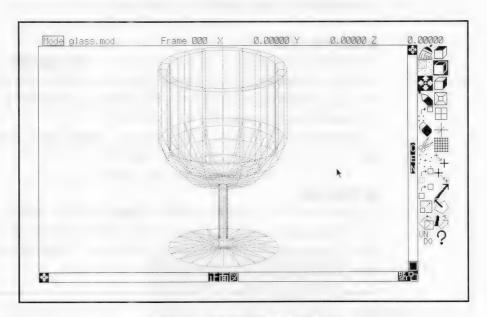
■点の作成と削除

座標平面内でマウスの左ボタンをクリックすると、マウスカーソル位置に点を打ちます。マウスカーソルを移動してクリックすると、クリックした位置に点が打たれて1つ前に作った点と線で結ばれます。このようにして PLANE コマンドを終了すると、最後に作った点と最初に作った点とが結ばれてポリゴンができます。

座標平面内でマウスの右ボタンをクリックすると、1番最後に作った点が削除されます。続けてクリックすると、順番に新しい点から削除されていきます。

■平面の選択

PLANE コマンドは、「側面図」(X 平面)、「上面図」(Y 平面)、「正面図」(Z 平面)のいずれかの平面上の原点を中心とし、1辺が2の正方形の領域で作業を行います。作業平面を変更するときは、作業したい平面の表示をマウスでクリックして反転表示させます。3つの平面はそれぞれポリゴンエディタの三面図に対応しています。

画面 5-20　PLANE 使用例(グラスの元データ)

PLANEコマンドで作成したポリゴンデータをROTコマンドを使って回転体にする場合は、ポリゴンがZ軸を中心に回転するため、「側面図」か「上面図」を選択してください。またTHICKコマンドを使って厚みを付ける場合には、Z軸方向に厚みが付くため「正面図」を選択してください。

画面5-21　PLANE、ROT使用例（グラス）

球を作成（SPHERE）

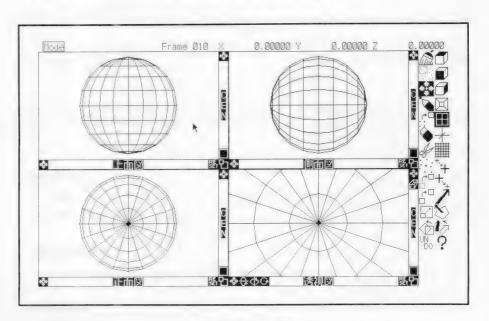
SPHEREコマンドを使ってみます。

ツリーエディタから SHIFT ＋ ! キーを押すと画面最下行に「実行するコマンド：」が表示されて入力を促しますので、次のように入力してみてください。

sphere　10 ↵

画面 5-22　球のデータ(SPHERE コマンドの実行結果)

この例では「分割数」を 10 にしたため、全部で 100(＝10×10)の面で球のデータが作られています。

続けて「ポリゴンエディタ」を選択すれば、球が表示されます。

画面 5-23　球の表示

このようにツールを使用すれば、幾何学的な物体データを簡単に作れます。

6章
モーションエディタ

モーションエディタは、ポリゴンを動かせるようにするエディタで、モーションファイルとパラメータファイルを作成します。モーションファイルとは、〝パラメータ″ファイルのファイル名とコマ(時刻)を示すファイルです。拡張子は〝.mot″を使用します。パラメータファイルとは、あるコマにおける変数の値を与えるファイルです。拡張子は〝.<コマ数>″を使用します。

モーションファイルを読み込んで編集するときは、次のようにMODEを起動します。

B>mode　－m　test.mot　sample.mod ↵	…	ベースネームがモデルファイルと違うとき
B>mode　sample ↵	…	ベースネームがモデルファイルと同じとき

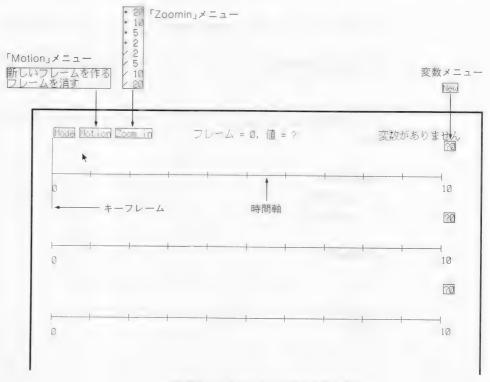
たとえば付属のsample.motとそのパラメータファイルの内容は、以下のようになっています。

```
sample.mot
0　sample.000
10　sample.010

sample.000
0

sample.010
90.1639
```

6.1　モーションエディタ画面

画面には、横に伸びる時間軸が3本と4種類のメニュー(「Mode」「Motion」「Zoom in」「?0」)が表示されています。

この時間軸1つが1つの変数に対応しており、時間軸の目盛り上にキーフレームを置いて、その目盛りに対応するコマにおける変数の値を指定します。時間軸上には、すべてのキーフレームを通る曲線が表示されます。この曲線が各コマにおける変数の値の推移を表しています。キーフレームの存在しないコマでも、この曲線の高さがそのコマの値となります。

それぞれの時間軸がどの変数を表しているのかは、時間軸の右上に表示されたメニューで指定します。変数の数は「?0」をドラッグして「New」の表示の上でマウスを放すと増やせます。表示する変数を切り替えるときもこのメニューで選択します。残りのメニューではエディタ間の移動、キーフレームの作成、削除および時間軸の縮尺の変更をします。

画面 6-1　モーションエディタの画面

6.2 モーションデザイン

「モデリング実践」で作成した「test.mod」を動かしてみましょう。

モーションデータを使って動かすためには、ツリーエディタでポリゴンのどこを動かすかを指定しなければなりません。

■変数の宣言

Y 軸にそって回転させるように設定してみます。

まず次のように入力して「test.mod」を読み込みます。

B>mode　test ⏎

ツリーエディタ画面で SHIFT ＋ O を入力して「NONE」を作ります。続けて C 、 R 、 Y 、 L 、 SHIFT ＋ $ と入力して、先頭の行に「ry　$0」を挿入します。すると「ry　$0：0」と表示されます("：0"は0コマの値)。

画面 6-2　変数の宣言

```
test.mod─┬─ry $0:0
         ├────ty -5
         │    ├─sx 4
         │    ├─sy 4
         │    ├─sz 4
         │    └─i pyramid.mod
         └─i floor.mod
```

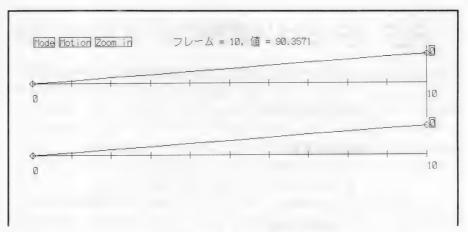
※モーションエディタではマイナスのつく変数名は指定できないので、マイナス値を入れないようにしてください。

■動きの指定

10 コマで約 90 度回転するように作ってみましょう。

「モーションエディタ」を選択して、「？0」メニューをマウスでクリックすると表示が「0」になり、10 コマ目に○印が表示されます。

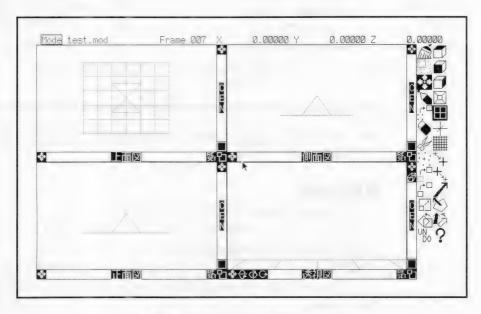
次に「10」の上でクリックして、キーフレームをフレーム 10 に移動します。そしてフレーム 10 の上にある○印を「値＝90」(小数点がついていても構いません)になるまで上方にドラッグします。

移動が終わったらフレーム 0 でクリックしてキーフレームを 0 コマ目にもどします。

画面 6-3　10 コマ目の指定

■動画表示

モーションの確認はポリゴンエディタで行います。

「ポリゴンエディタ」を選択すると、画面上部に「Frame 000」のように表示されています。

ここで スペース キーを押すと、1 コマずつ画面が送られていきます。

画面 6-4　ポリゴンエディタ画面(7 コマ目)

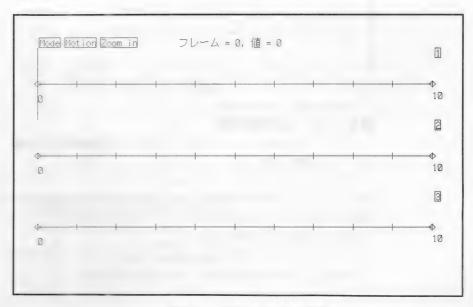
6.3 モーションデータの作成と確認

モーションエディタによるモーションデータの作成は、変数の宣言、キーフレームの作成、キーフレーム値の決定といった手順になります。これらの処理は、おもにマウス操作で行います。

モーションデータの作成

■変数の宣言

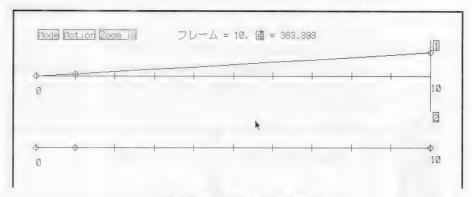
モーションエディタでは、まず変数の使用を宣言しなければなりません。最初に起動した状態では変数は1つも宣言されていないので、それぞれの時間軸の右上に表示された「?0」をクリックして変数を宣言します。マウスでドラッグすると「New」と表示、放すと「0」と表示され、時間軸の0コマ目と10コマ目に○印でキーフレームが表示されます。このメニューは、この時間軸が変数$0のデータを表示していることを示し、またキーフレームの値も0を示しています。

次に、「0」と表示されたメニュー上でドラッグすると、メニューの下に「New」と表示されます。このままマウスを移動させ、「New」の上でボタンを放すと今度は「1」と表示され、変数$1が宣言されます。以下同様に、$2、$3とモデルデータ内で使用しているだけの変数を宣言します。

画面6-5　変数$1、$2、$3を宣言

■キーフレームの作成と削除

変数を宣言すると自動的に始点と終点のキーフレームが作成されますが、キーフレームを1つ追加してみます。まず、作成したいコマ上でクリックしてモーションカーソル(｜印)を移動させます。次に「Motion」上でドラッグし、そのまま移動させて「新しいフレームを作る」を選択してボタンを放すと、モーションカーソル位置に新しいキーフレームが表示されます。

作成したキーフレームの初期値は、すでに作成されたキーフレームを結んだ線上、またはその延長線上になります。これを利用すれば、任意の値を持つキーフレームが簡単に作れます。たとえば、0コマ目に0、1コマ目に36の値を持つキーフレームを作ります。次に10コマ目にキーフレームを作るとそのキーフレームの初期値は360となります。このようにすれば、大きな値を容易に設定できます。

キーフレームを削除するには、カーソルを削除したいキーフレーム上に移動させ、作成したときと同様に「Motion」から「フレームを消す」を選択します。

画面6-6　1コマ目に36を指定

■キーフレームの値の変更

キーフレームの値(キーフレームのコマの変数の値)は時間軸上が0で、上がプラス、下がマイナスを表します。

まずマウスカーソルをキーフレームの○印上でドラッグしてモーションカーソルを移動させ、これを上下いずれかにドラッグして移動します。それぞれ○印がマウスに合わせて移動し、マウスボタンを放すとキーフレームの値が変更されます。このときのキーフレームの値は、画面最上行に表示されます。

キーフレームは左右に移動させればそれぞれのキーフレームが指すコマを変更できます。変更は上下方向の移動と同様にドラッグして行います。

■時間軸の縮尺の変更

初期状態では、1画面には0から10までのコマが表示されていますが、コマ数の範囲は変更できます。「Zoom in」メニューで任意の倍率を選択します。"*"を選択すると時間軸の目盛りの間隔が拡大、"/"を選択すると縮小します。

モーションデータの確認

モーションエディタで作ったモーションデータは、ポリゴンエディタで表示の確認ができます。モデルファイルに変数が使用されている場合は、ポリゴンエディタで表示するときの変数の値にモーションエディタのカーソルが指し示すコマでの値が使用されます。モーションエディタでカーソルを任意のコマに移動させることで、そのコマでどのような画像が得られるかが分かります。

また、ポリゴンエディタからでも以下のキーを使用して、表示するコマを指定できます。

キー操作	内　容
[N]	次のコマを表示
[スペース]	次のコマを表示
[P]	前のコマを表示
[CTRL]+[H]	前のコマを表示

なお、モーションデータがないときに[スペース]を押すと、画面を再描画します。

6.4　キー操作

モーションエディタではマウスによる操作を次のキー操作で代用できます。

■カーソル移動

カーソルを移動させるキー操作はツリーエディタと同じです。

キー操作	内　容
[H]	カーソルを左に移動
[J]	カーソルを下の時間軸に移動
[K]	カーソルを上の時間軸に移動
[L]	カーソルを右に移動

■画面の操作

キー操作	内　容
SHIFT + H	時間軸の表示範囲を左に移動
SHIFT + L	時間軸の表示範囲を右に移動
CTRL + L	画面の再描画

■表示変数の変更

キー操作	内　容
SHIFT + J	カーソル位置の時間軸で、現在表示している変数の次の変数を表示
SHIFT + K	カーソル位置の時間軸で、現在表示している変数の前の変数を表示

■縮尺の変更

キー操作	内　容
*	時間軸の目盛りの間隔を拡大。続けて実行すれば、さらに拡大
/	時間軸の目盛りの間隔を縮小。続けて実行すれば、さらに縮小

■その他の操作

キー操作	内　容
W	モーションデータをファイルに保存
SHIFT + ?	説明を表示

7章 ファイルの切り替え

MODEでは複数のモデルデータファイルを読み込めますが、同時には編集できません。実際に編集したいモデルファイルを「ファイルの切り替え」で指定します。

モデルファイルの切り替えはツリーエディタでもできますが、複数のファイルを読み込んでいる場合は、「ファイルの切り替え」のほうが目標とするモデルファイルに早く到達できます。

7.1 ファイルの切り替え画面

たとえば「モデリング実践」で作成した「test.mod」を読み込んで「Mode」メニューで「ファイルの切り替え」を選択すると、次の画面が表示されます。

```
Mode

ファイル名                                        エラー 編集中
------------------------------------------------- ------ ------
test.mod                                              no    yes
pyramid.mod                                           no     no
floor.mod                                             no     no
floorsub.mod                                          no     no
```

画面7-1　ファイルの切り替え画面(test.mod)

画面には、いま読み込んでいるモデルデータファイル一覧とそれぞれのファイルの状態が表示されます。ファイルの状態とは、エラーがないか、読み込まれた後に編集されたかの2つです。

7.2 キー操作

「ファイルの切り替え」ではツリーエディタと同様にキーボードで操作をします。

ファイルを切り替えるときは、カーソルを編集したいファイルの前に移動させ、スペースキーを押します。ファイルを選択するとそのファイルを読み込み、ツリーエディタへ移動します。カーソルの移動の方法はツリーエディタとほぼ同じで、次のようになっています。

■カーソル移動

キー操作	内　容
CTRL+B	1画面上に移動
CTRL+U	半画面上に移動
CTRL+F	1画面下に移動
CTRL+D	半画面下に移動
J	1つ下へ移動
K	1つ上へ移動

■その他の操作

その他の機能として、次のものがあります。

キー操作	内　容
R	ファイル名の変更
スペース	ファイルを確定し、ツリーエディタへ移動
CTRL+L	画面を再描画
SHIFT+?	ファイルの切り替えの説明を表示

7.3 ファイルの切り替えの実際

それでは実際に「ファイルの切り替え」を使ってみましょう。

次のようにして「sample.mod」を読み込み、「Mode」メニューで「ファイルの切り替え」を選択(または[NFER]+[F])してください。

B>mode　sample [↵]

```
Mode

ファイル名                                                エラー 編集中
--------------------------------------------------------  ------ ------
sample.mod                                                  no     no
sphere.mod                                                  no     no
cube.mod                                                    no     no
pyramid.mod                                                 no     no
floor.mod                                                   no     no
```

画面 7-2　ファイルの切り替え画面

[J]キーで「floor.mod」にカーソルを移動して[スペース]キーを押すと、ツリーエディタに移動して「floor.mod」の内容が表示されます。

ここで「ポリゴンエディタ」を選択すれば「floor.mod」のワイヤフレームが表示されます。

8章
モデリング実践

この章では実際のモデリングの例を示して、どのような手順で物体を作り上げていくのかを説明します。

8.1 効果的なモデリング

能率良く、きれいなポリゴンを作るために、次のことを念頭においてモデリングをしましょう。

- ●モデリングする前にラフスケッチを用意
- ●最初は大まかに作成
- ●基本は「正面図」
- ●ツールを利用
- ●部品ごとにブロック化

最初は誰でもポリゴンエディタだけを使ってモデリングしてみるでしょう。しかし、イメージどおりに形を作るのはなかなかむずかしいものです。そこで、次のような手順を踏むことをお勧めします。

まず下準備として、紙にラフスケッチを描きます。頭の中でイメージしただけでは全体のバランスがとりにくかったりするからです。

次にポリゴンエディタを使って実際にモデリングを行います。ただし、最初から完成したものを作るのではなく、ポリゴンの点の数は少な目に粗く作ります。後で点を減らすのは面倒ですが、増やすのは簡単だからです。

また点の座標は、きりのよい値にしましょう。きりのよい座標ならば軸に平行なポリゴンを作りやすいし、後で点を移動させるのも簡単だからです。

そして「正面図」で作業します。XY 平面内だけでならば形もとりやすいし、Z 軸方向には後でツール（THICK コマンド）を使って厚みを付けることができるからです。

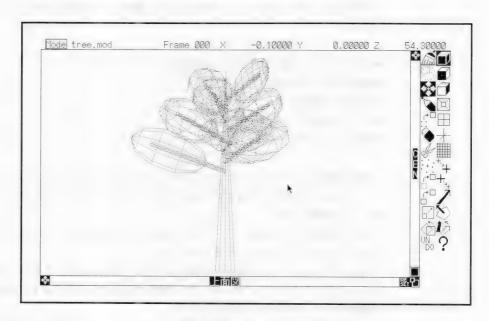
ある程度形がとれてきたら細部を調整します。頂点の数を増やして曲線に近く見せるためには、〝中点作成〟の機能を使います。作った点は曲線となるような位置に移動させて使います。

以上のように、まず粗く作って全体の形がとれてから細部を作るようにします。粗いほうがポリゴンエディタでの描画が速くなるし、先に全体の形がとれたほうが後で大きな変更をしなくてすみます。

ツリーエディタのｓコマンドを使って変形させたり、モーションエディタで部分的に動かせるように部品ごとにブロックに(階層化)しておきましょう。

また、球や円錐などのポリゴンは１から作るとたいへんなので、ツールを利用しましょう。

8.2 木の作成 －Step 1－

おもにツールとツリーエディタを使用して木を作ってみます。

画面 8-1　tree.mod(完成例)

幹、枝、葉に分割

まず木をどのようなブロックに分けて構成するかを考えます。木は幹、枝、葉に分けられます。また枝も細い枝のブロックと太い枝のブロックに分かれます。

次にこれらのブロックをどのような物体で見立てるかを考えます。幹は先端にいくほど細くなりますから、細長い円錐で表します。

円錐を作るツールはCONEです。枝は幹に比べれば太さの変化がありませんから、円柱で現します。葉は1つずつ作っていたのでは計算量が増えるだけですから、1つの枝に付く葉をまとめて球で現すことにします。

ツールで部品を作成

ブロックの構成が決まったら、それぞれのブロックのポリゴンデータを作ります。このとき、必要以上に細かいポリゴンデータを作らないように注意しましょう。部品の配置は、ポリゴンエディタでワイヤフレームを表示させて確認します。

ポリゴンエディタで表示するとき、ポリゴンの数が多いと表示し終わるまでに時間がかかってしまいます。そこで、最初は点の数が少ないポリゴンデータを使い、配置が終わったらポリゴンデータを差し替えて完成させます。ツールを使うとき、最初は分割数に小さな値を指定し、最後にイメージしていた分割数を指定します。

ここでは次のようにそれぞれ分割数を5にしてデータファイルを作ります。

```
B>cone  5  >  cone.mod ⏎         … 幹のデータ
B>tube  1  1  5  >  tube.mod ⏎   … 枝のデータ
B>sphere  5  >  sphere.mod ⏎     … 葉のデータ
```

以下では、この3つのデータファイルを使った組み立て方を説明します。

ポリゴンデータの組み合わせ

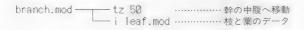
最後に作ったポリゴンデータをr、t、s、iコマンドを使って組み合わせます。それぞれのコマンドの引数は、ポリゴンエディタでワイヤフレームを確認しながら決めました。

まず最初に大きさを決めるために幹のデータ「cone.mod」をsコマンドを使って加工します。幹のポリゴンは底面をXY平面に持つ高さ1、半径1の円錐です。これを次のようなコマンドを使って高さ100、太さ5のデータに加工します。

画面8-2　幹

加工が終わったら「tree.mod」という名前を付けておきましょう。

次にこの幹の大きさに合わせて枝を取り付けます。とりあえず幹の中ほどに1本取り付けてみましょう。「branch.mod」というファイルを次のような内容で作ります。新たにファイルを作るには、ツリーエディタのeコマンドを使用してファイル名を指定します。

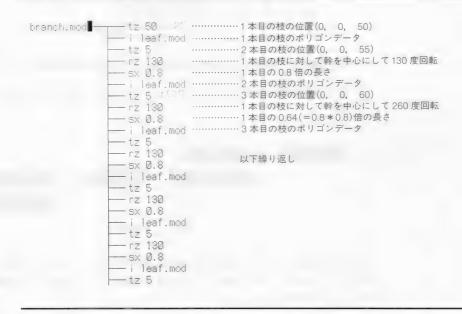
画面8-3　branch.mod

そして、「ファイルの切り替え」で「leaf.mod」を選択して、「tube.mod」を加工します。枝のデータ「tube.mod」は、底面をXY平面に持つ高さ1、半径1の円柱なので、これもrコマンドとsコマンドを使って加工します。

画面8-4　leaf.mod

続いて「ファイルの切り替え」で「tree.mod」を選択し、次のように「i　branch.mod」を追加します。

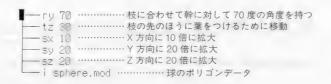
画面 8-5　tree.mod

これで幹に1本の枝が付きました。続けてもっとたくさんの枝を付けましょう。「ファイルの切り替え」で「branch.mod」を選択して加工します。これらの枝は、幹(Z軸)を軸にして回転させながら取り付けていきます。回転の角度は、1周したときにずれるように130度とします。また、枝は上に行くほど短かくなるようにします。これをデータにすると、以下のようになります。

画面 8-6　branch.mod

最後に枝に葉を付けます。葉のデータ「sphere.mod」は半径1の球のポリゴンデータです。これをX軸方向につぶれた形に拡大して、それぞれの枝に取り付けます。「ファイルの切り替え」で「leaf.mod」を選択して**画面8-7**の内容を追加してください。

「sphere.mod」のようなデータを作るときは、順番に注意する必要があります。r、t、sコマンドは、コマンドよりも後ろにあるポリゴンデータに影響を与え、ポリゴンデータに近いほうから作用することは説明しました。1行目と2行目をみると、葉のポリゴンデータはZ軸方向に30移動してから、Y軸を中心に70度回転していることが分かります。このコマンドによって、葉のポリゴンのX座標はマイナスになります。

しかし、このコマンドを逆の順序で実行したらどうなるでしょうか。まずY軸を中心にして回転します。初め葉のポリゴンは原点を中心にしていますから、位置は変わりません。次にZ方向に30移動しますから、結局X座標は0のままで、葉のポリゴンは幹に埋もれたままです。

画面8-7　葉

ここまでに使用したモデルファイル中のiコマンドのみを取り出すと次のようになります。

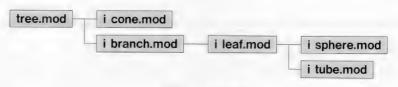



図8-1　tree.mod

組み立てが完成したので、今度はそれぞれのポリゴンデータをもっと細かいものに取り替えます。とはいえ、あまり細かくしすぎると画像ファイルを作るときに時間がかかりますから、ほどほどにしましょう。

作例の表示

それでは完成例を表示してみましょう。インストールプログラムではコピーされないので、次のような準備をしてください。

①「実行用システムディスク」をドライブAに入れてMS-DOSを起動(以下の手順はフロッピーディスクのみのシステム構成)。
②初期化済みのディスクをドライブBに入れる(初期化の手順は「インストール」を参照)。
③「実行用システムディスク」を「保存用システムディスク」と入れ換える。
④次のように入力。

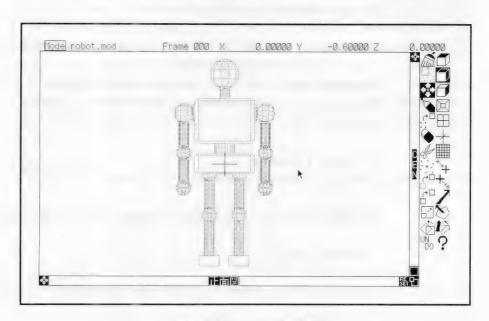
```
B>copy a:¥tree b:⏎
B>copy a:¥color b:⏎
```

⑤「保存用システムディスク」を「実行用システムディスク」と入れ換え、次のように入力。

```
B>mode tree⏎
```

「ポリゴンエディタ」で画面いっぱいに表示させたのが**画面8-1**です。

8.3 ロボットの作成　-Step 1-

次にロボットを作ってみましょう。手足を動かして歩くようにデザインします。

画面 8-8　robot.mod（完成例）

頭、胴体、手足に分割

まず、ロボットをどのようなブロックに分けて構成するかを考えます。ロボットは頭、胴体、右手、左手、右足、左足に分けられます。次にこれらのブロックをどのような図形で作るかを考えます。簡略化するためにどのブロックも基本的に円柱、球、直方体で作ります。手足の関節などの動かす部分は動きをきれいにするために球や半球にしておきます。胴体はCUBEコマンドで直方体の角を削って作ります。

ツールで部品を作成

ブロックの構成が決まったら、それぞれのブロックを作成するのに必要なポリゴンデータのファイルを作ります。ここでは次のようなデータファイルを作りました。

「sph－h.mod」(半径1の半球)と「cube1.mod」(1辺20の立方体)は部品集にあるものを使用し、後はツールを使って作ります。

```
B>sphere  4  >sph.mod ↵                    … 半径1の球
B>tube  -t  1  10  10  >tube.mod ↵         … 底面の半径1、高さ10の円筒形
B>cube  12  1  >cube2.mod ↵                … 1辺12の角を削った立方体
```

大きさを決定

ポリゴンデータを組み合わせる前にどのくらいの大きさのロボットを作るのか、おおざっぱに決めます。たとえば次のようにします。

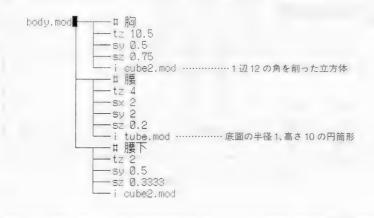
全長	＝40
足	＝20
胴体	＝15
頭と首	＝5

各ブロックを作成

■胴体を作成

身体の中心である胴体から作ります。大きさは前項で決めたように15くらいで、また形としては腰を円柱、その上下を「cube2.mod」の角を削った立方体で表すことにします。

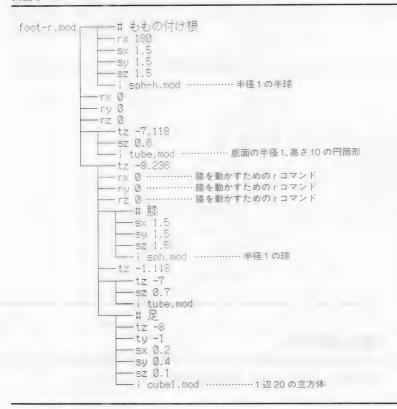
最初にツリーエディタでt、s、iコマンドを使って「cube2.mod」と「tube.mod」を自分がイメージしたように結合します。次にポリゴンエディタで表示してバランスを確かめます。これを何度も繰り返してコマンドの引数を決めます。このようにしてできたファイルが「body.mod」です。

画面 8-9　body.mod

■足を作成

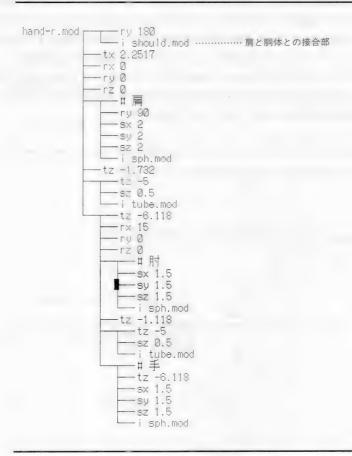
まず、後から動かす部分を決めます。ここではももの付け根と膝を動かすことにします。この2つの部分はrコマンドの引数を変数にして動かすので、球または半球を使うとよいでしょう。その他の部分は足を直方体で、ももとすねを円筒形で表します。

足を作る場合に重要なのは変数の値をいろいろ変えても足全体が足の形を成しているということです。ここがロボットを作る上で最もむずかしいところになります。

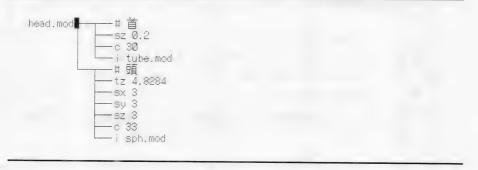
膝を例に説明すると、まず膝を構成する球の中心が原点にくるように膝から下を作ります。次にその部分を1つのブロックと考えてブロック全体をいつでも原点中心に回転させられるようにrコマンド(rx、ry、rz)を挿入しておきます。引数は0でも構いません。最後にrコマンドを含めた全体を膝から上との結合点へ移動します。同様にももの付け根も作ります。ももの付け根は「sph－h.mod」(半径1の半球)、膝は「sph.mod」(半径1の球)、足は「cube1.mod」(1辺20の立方体)をそれぞれ加工して作り、ももの付け根と膝、足の間は「tube.mod」(底面の半径1、高さ10の円筒形)で結びます。

画面8-10　foot－r.mod

■腕を作成

腕も足と同様に作成します。肘と肩を動かしたいので、これらの部分と手は球で表します。その他の部分は円筒を使います。ここで重要なことは肩と胴体との接合部をどうするかということで、腕を動かしてもきれいに見えるように考えてみてください。ここでは「should.mod」という別のファイルにしています。

画面 8-11　hand−r.mod

■首と頭を作成

ここまでにできた部品とのバランスを考えて首と頭を作ります。とりあえず首を円筒、頭を球にしてありますが、頭は自分で作ってみてください。

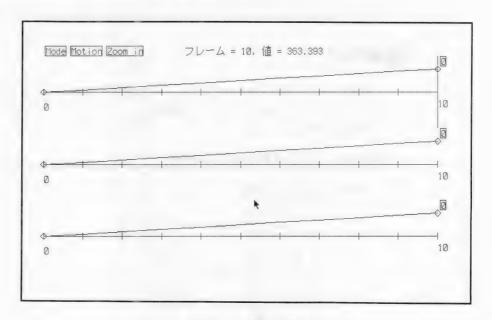
画面 8-12　head.mod

組み立て

最後にすべての部品を組み立ててロボットを完成させます。ファイル名を「robot.mod」にして i コマンドで各ポリゴンデータを読み込み、ポリゴンエディタで各パーツのバランスを見ながら組み立てます。

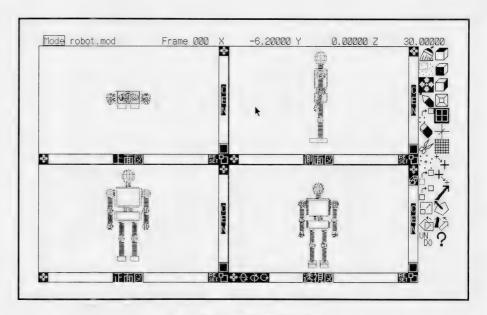
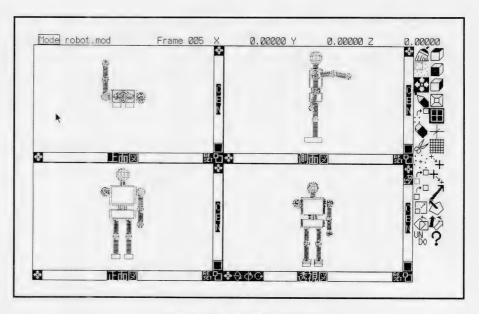
画面 8-13　ファイル名を「robot.mod」に指定

動きを付加

これで一応ロボットができましたので、今度はモーションエディタを使って動かしてみます。では右腕をX方向を軸に回してみましょう。

まず最初に肩を成す球のところにあらかじめ入れておいたrコマンドのうちrxの引数を変数$0に置き換えます。次にモーションエディタで「フレーム」と「値」の推移を指定します。フレーム数と値の大きさは、ポリゴンエディタで表示して確認しながら決めます。ここでは「フレーム＝10」、「値＝360」に指定します。

画面8-14　動きの付加

それでは、ポリゴンエディタで表示して動きを見てみましょう。スペースキーを押して次のフレームを表示させていくと、ロボットの右腕が少しずつ回転していきます。

このようにロボットにいろいろな動きをさせてみてください。

画面 8-15　腕の回転(0 コマ目)

画面 8-16　腕の回転(5 コマ目)

作例の表示

それでは完成例を表示してみましょう。インストールプログラムではコピーされないので、次のような準備をしてください。

①「実行用システムディスク」をドライブAに入れてMS-DOSを起動(以下の手順はフロッピーディスクのみのシステム構成)。
②初期化済みのディスクをドライブBに入れる(初期化の手順は「インストール」を参照)。
③「実行用システムディスク」を「保存用システムディスク」と入れ換える。
④次のように入力。

```
B>copy　a:¥robot　b:[↵]
B>copy　a:¥color　b:[↵]
```

⑥「保存用システムディスク」を「実行用システムディスク」と入れ換え、次のように入力。

```
B>mode　robot[↵]
```

「ポリゴンエディタ」で画面いっぱいに表示させたのが**画面8-8**です。[スペース]キーを押すと、次のフレームが表示されます。

また、mode　robot2[↵]と入力して「ポリゴンエディタ」を選択すれば、横向きのロボットが表示されます。

Part 3
シェーディング

Part 2ではワイヤフレームの画像作成までを解説しましたが、
Part 3ではカラー画像を表示するための
コマンドデータについて説明します。
9章では色指定の方法について説明し、
10章では8章で作った2つの作品に色を付けます。

9章 色情報

物体や背景に色を付けるにはツリーエディタ上でいくつかの指定をしなければなりません。ここでは色指定に必要な視野モデル用コマンド、背景色の指定コマンド、光源モデル用コマンド、表面モデル用コマンド、シェーディングパラメータ指定用コマンドについて説明します。また、簡単に背景色を作成できるカラーエディタについても解説します。

9.1 色を構成する要素

カラーで画像を表示するためには、かならず次の指定をしなければなりません。

①横方向の解像度、視野の角度 … 視野モデル用コマンドで指定
②縦方向の解像度、視野の角度 … 視野モデル用コマンドで指定
③背景色 … 光源モデル用コマンドで指定
④光源色 … 光源モデル用コマンドで指定
⑤光源の向き … シェーディングパラメータ指定用コマンドで指定
⑥光源 … シェーディングパラメータ指定用コマンドで指定
⑦物体色 … 表面モデル用コマンドで指定

第2編で説明した「sample.mod」でも**画面 9-1** のように指定しています。

画面 9-1 「sample.mod」の色指定

```
sample.mod ┬─x 213,64,0,1;    ………… 横方向の解像度を 213 ドット、視野の角度を 64 度に指定
           ├─y 133,40,0,1;    ………… 縦方向の解像度を 133 ドット、視野の角度を 40 度に指定
           ├─l 0,0.5,0.5,0.5 ………… 背景色を指定（灰色）
           ├─l 1,1,1,1        …………… 光源色を指定（白色）
           ├─b 0,(0,0,0)      …………… 光源の向きを指定
           └─b 1,(0,5,-1)     …………… 光源を指定
```

以下では各コマンドごとにその使い方を紹介します。

9.2　視野モデル用コマンド

書　式

x　〈横方向の解像度〉,〈視野の角度〉[,〈分割した画像の位置〉,〈分割数〉]
y　〈縦方向の解像度〉,〈視野の角度〉[,〈分割した画像の位置〉,〈分割数〉]

例

x　640，40　　　　… 640×400 ドットで視野角 40 度に指定
y　400，40

x　512，50，0，1　　… 3 分割する場合の最初の画像
y　 80，40，0，3

x　512，50，0，1　　… 3 分割する場合の最後の画像
y　 80，40，2，3

視野モデル用コマンドは、生成する画像の解像度(ドット数)と視野の角度を指定します。このコマンドによってクリッピングや透視変換の設定が変更されます。これらのデータは画像の生成にかならず必要なものですから、モデルファイルの先頭に 1 回だけ指定します。

画像の解像度は、出力機器に合わせた値(本パッケージでは最大 640×400 ドット)を整数値で指定します。視野の角度は、単位を度で表した視野の広さのことで、実数で指定します。大きければ(上限は 180 度)広角レンズのような効果を生み、小さければ望遠レンズのような効果を生みます。通常は 40 度ぐらいを指定しておけば、自然な画像が得られます。

また視野モデル用コマンドは、画像を分割して生成するための指定も行います。MODE システムではすべてメモリ上で処理するため、大きな画像を生成する場合メモリ不足となることがあります。そういうときは画像を分割して生成し、後から結合します。画像の分割は縦と横の両方向に可能ですが、横方向に分割した画像を結合する方法がないので、通常は縦方向の分割を利用します。ただし、画像の一部を見るときは横方向の分割を利用できます。

画像を n 個に分割して生成する場合は、画像の位置は 0 から n−1 の値をとります。つまり、画像を生成するごとに画像の位置を 0 から 1、2、3 と書き換えて、n 個のイメージファイルを生成します。生成したファイルの結合は MS-DOS の COPY コマンドを使って次のようにします。

```
B>copy  /b  image1+image2+image3  image ↵
```

これで image1、image2、image3 の 3 つのファイルが結合され、image という 1 つのファイルになります。

9.3 シェーディング用コマンド

シェーディング用コマンドには、光源モデル用コマンド(l)、表面モデル用コマンド(c)、シェーディングパラメータ指定用コマンド(b)があります。

光源モデル用コマンド

書　式

l　〈光源番号〉,〈赤〉,〈青〉,〈緑〉
b　〈シェーディング情報番号〉,(X,Y,Z)

例

l　0，0，0，0	… 背景色を黒に指定
l　1，1，1，0	… 光源色を黄に指定
b　0，(0，0，0)	… 光源の向きを指定
b　1，(1，1，1)	… 光源を平行光源に指定

〝l〞コマンドは、シェーディングの際に必要となる色情報を光の3原色で指定します。赤(Red)、青(Blue)、緑(Green)の割合を0から1までの実数で記述します。

どのような色情報が必要になるかは、後述する表面モデル用コマンド(c)で指定されます。〝c〞コマンドによって光源番号とその光源の種類(平行光源や点光源など)が指定されます。

たとえば、ほとんどのシェーディング方式(〝c〞コマンドによって指定される)で光源番号1番が平行光源として使用されます。この平行光源のベクトルを指定するのに〝b〞コマンドを使用します。ベクトルは、シェーディング情報番号1番で設定された座標から0番で設定された座標の向きに指定されます。

ただし、光源番号0番(l　0)だけは例外で、シェーディング方式に関係なく常に背景の色情報を設定するので、〝b〞コマンドは不要です。

表9-1にシェーディング番号とシェーディング方式の対応を示します。

シェーディング番号	意　　味	必要な情報
0	白一色の塗りつぶし	なし
1	白色のシェーディング（ランバートモデル）	光源番号1番　シェーディング情報番号0番、1番（1番から0番の向きの平行光源）
2	白色のフォンシェーディング	光源番号1番　シェーディング情報番号0番、1番
3	白色のブリンシェーディング	光源番号1番　シェーディング情報番号0番、1番
4	黄色のプラスチック風	光源番号1番　シェーディング情報番号0番、1番
5	黄金色の金属風	光源番号1番　シェーディング情報番号0番、1番
6	2つの点光源に照らされた白色のプラスチック	光源番号1番、2番（点光源Ⓐ）、3番（点光源Ⓑ）　シェーディング情報番号0番、1番、2番（点光源Ⓐの位置）、4番（点光源Ⓑの位置）
7	高度差によって変化する色のシェーディング（ランバートモデル）	光源番号1番　シェーディング情報番号0番、1番、6番（球の中心）、7番（球面上の点）
8	青色のランバートモデルとバンプマッピング	光源番号1番　シェーディング情報番号0番、1番、8番（波紋の中心1）、9番（波の到達点1）、10番（波紋の中心2）、11番（波の到達点2）
9	木目のソリッドテクスチャとランバートモデル	光源番号1番　シェーディング情報番号0番、1番、12番（円の中心）、13番（円弧上の点）、14番（円弧上の点）
10	画像のマッピング（ソリッドテクスチャ）とランバートモデル	光源番号1番　シェーディング情報番号0番、1番、15番（画像平面の原点）、16番（画像平面のX軸上の点）、17番（画像平面のY軸上の点）　画像ファイル〝map.pic〟
11	グラデーションのリフレクションマッピングとブリンシェーディング	光源番号1番、4番（天頂の色）、5番（地平線のすぐ上の色）、6番（地平線のすぐ下の色）、7番（天頂と反対側の色）シェーディング情報番号0番、1番、18番、19番（18番から19番の向きが天頂）
12	画像のリフラクション（屈折）マッピング	光源番号1番　シェーディング情報番号0番、1番、20番（画像平面の原点）、21番（画像平面のX軸上の点）、22番（画像平面のY軸上の点）　画像ファイル〝map.pic〟
20〜29	フォンシェーディング	光源番号1番　シェーディング情報番号0番、1番　シェーディングパラメータファイル〝Shade.20〟〜〝Shade.29〟
30〜39	ブリンシェーディング	光源番号1番　シェーディング情報番号0番、1番　シェーディングパラメータファイル〝Shade.30〟〜〝Shade.39〟

表9-1　シェーディング番号に対応したシェーディング方式

ランバートモデルとは最も簡単なシェーディングモデルで、ランバート(Lambert)の反射則「入射した光はすべての方向に等しく拡散する」を用いて光のアンビエント成分とデフューズ成分だけを計算するものです。

フォンシェーディングとはスペキュラー成分を取り入れた初期のモデルで、物体の表面のざらつきまで計算に入れるものです。ただし、表面の粗さを自然数 n で決めるため、細かな調節ができません。

ブリンシェーディングとは、「物体の表面を細かく見れば実際にはランダムな方向を向いた小さな面の集合からなる」という考えのもとに作られたシェーディングモデルです。ブリンのシェーディングモデルは、フォンのシェーディングモデルよりも多くの計算時間を必要としますが、鋭いスペキュラーを持つ物体や浅い角度から入射してくる光もリアルに表現できます。

それぞれのシェーディングモデルの計算式が必要な方は『応用グラフィックス』を参照してください。

添付のディスクには以下の 10 色の背景色ファイルが用意されているので、使ってみましょう。

ツリーエディタで読み込むときは i コマンドを使って次のように指定します。

i 〈ファイル名〉

ファイル名	色
YELLOW.CLR	黄
GREEN.CLR	緑
SKYBLUE.CLR	水色
BLUE.CLR	青
PURPLE.CLR	紫
PINK.CLR	ピンク
RED.CLR	赤
WHITE.CLR	白
GRAY.CLR	灰色
BLACK.CLR	黒

これらのファイルの内容は以下のようになっています(2 行目は共通)。

1 行目： 1 0, 〈赤〉, 〈青〉, 〈緑〉
2 行目： 1 1, 1, 1, 1

表面モデル用コマンド

書　式　c　〈シェーディング番号〉
b　〈シェーディング情報番号〉,（X，Y，Z）

物体の表面に色を付けるには、"c"コマンドを使用します。色指定の方法は、あらかじめ用意された色の中から選び、その色に付けられたシェーディング番号を記述します。このコマンドで色が指定されると、次に"c"コマンドが指定されるか、またはブロックの最後まで有効です。つまり、同じ色の物体ごとにブロックにして、そのブロックの内部で"c"コマンドを使用すれば物体ごとの色指定が簡単にできます。

色指定と述べましたが、"c"コマンドは正確にはシェーディング方式を指定します。つまり色だけでなく、シェーディング方式やマッピングなども同時に指定します。このため、任意の色を指定するのは光源色の指定に比べて複雑です。

シェーディング番号の20〜39がユーザー定義用に用意されています。20〜29がフォンシェーディング、30〜39がブリンシェーディングです。いずれの場合でも"shade.〈シェーディング番号〉"（たとえば"shade. 20"）というファイルを次のような内容で作成します。

書　式

```
ar   ag   ab
dr   dg   db
sr   sg   sb
ss
```

例

```
 63   63   63
128  128  128
 64   64   64
  6
```

ar、ag、abは、物体の環境光（アンビエント）成分としてrgbの値を指定したものです。dr、dg、dbは、物体の拡散反射光（デフューズ）成分としてrgbの値を指定したものです。sr、sg、sbは、物体の鏡面反射光（スペキュラー）成分としてrgbの値を指定したものです。ssはスペキュラーの鋭さを表します。rgbの値は0から255の範囲の整数で指定します。ssはフォンシェーディングの場合は正の整数、ブリンシェーディングの場合は正の実数で指定します。

環境光成分とは、物体上の光が直接当たらない部分の色です。物体の回りの物からの反射光などによって、直接光が当たらない部分にも色が付きます。

拡散反射光成分とは、物体本来の色のことです。赤い物体にしたければ、この拡散反射光成分のrgbの値を赤にします。

鏡面反射光成分とは、物体上で光を反射した部分の色のことです。ハイライトとかテカりなどといいます。表面が滑らかな物体では、物体の表面上のごくわずかの部分だけが鋭く光輝きます。一方、表面が粗い物体では、反射は鈍くなります。これを指定するのが、スペキュラーの鋭さの値です。ブリンシェーディングの場合は、スペキュラーの鋭さの値が小さくなればなるほど光の反射の面積が小さくなり、大きくなればなるほど光の反射面積が大きくなります。フォンシェーディングの場合はこの逆です。

添付のディスクには以下の10色の物体色ファイルが用意されているので、使ってみましょう。

ファイル名	色
SHADE.30	黄
SHADE.31	緑
SHADE.32	水色
SHADE.33	青
SHADE.34	紫
SHADE.35	ピンク
SHADE.36	赤
SHADE.37	白
SHADE.38	灰色
SHADE.39	黒

指定は、次のようにします。

c 〈シェーディング番号〉

これらのファイルの内容は以下のようになっています(3、4行目は共通)。

1行目：　ar　ag　ab
2行目：　dr　dg　db
3行目：　128　128　128
4行目：　0.1

9.4 画像表示

たとえば mode　sample [↵] を実行して、3章で使ったサンプルファイルを読み込み、背景色や物体色を変えてみてください。

視野モデル用コマンド、光源モデル用コマンド、表面モデル用コマンドの指定が終わったら、画像を表示してみましょう。

「Mode」メニューの「画像表示」を選択する（または [NFER] ＋ [G] ）と、「recalc all animation (y/n)?」と聞いてきます。[Y] [↵] を押すと、しばらくしてから画面にカラーの画像が表示されます。

「画像表示」を終了するときは [Q] を押します。そうすると MODE を終了して MS-DOS のコマンドラインにもどります。

画面上に残っている画像を消すときは、次のコマンドを実行します。

B>cls3 [↵]

視野モデル用コマンドで、解像度を小さくすれば、画像の表示速度を速くできます。

※実際にカラー画像が表示されるまでには sample.mod で約10分ほどかかります（PC-9801VX を使用）。

9.5 画像連続表示

「Mode」メニューで「画像連続表示」を選択する（または [NFER] ＋ [A] ）と、「recalc all animation (y/n)?」と聞いてきます。[Y] [↵] を押すと、しばらくしてから画面にカラーの画像が1枚ずつゆっくりと表示されます。

「画像連続表示」を終了させるときも「画像表示」と同様に [Q] を押します。そうすると MODE を終了して MS-DOS のコマンドラインにもどります。

画面上に残っている画像を消すときは「画像表示」と同様に cls3 [↵] を実行します。

※カラー画像が表示されるまでには sample.mod で1画面あたり約10分ほどかかります（PC-9801VX を使用）。

9.6 カラーエディタ

MODE では色を表現する方法として RGB 方式を使用していますが、数値を指定してイメージどおりの色を作ることは非常に困難です。そこで画面を見ながら色作成ができる CEDIT というカラーエディタがあります(口絵㉑参照)。

RGB方式とHSV方式

CEDIT では RGB 方式と HSV 方式の 2 通りの方式で色を作り、〝l　n, r, g, b〞の形でデータを出力します。

■RGB 方式

RGB 方式では、色を R(赤)、G(緑)、B(青)の 3 色(いわゆる光の 3 原色)で表現します。色はこの 3 色それぞれの含まれる割合によって決まります。CEDIT では白色を R＝G＝B＝1.0 と定めています。

■HSV 方式

HSV 方式は人間の感覚をもとにした表現方法です。この方式では色を H(Hue)、S(Sturation)、V(Value)の 3 つの要素で表現します。H、S、V のそれぞれの意味は次のとおりです。

H 「青」「黄緑」などの色相を表す
S 「白っぽい」「灰色っぽい」などの色の飽和度を表す
V 「明るい」「暗い」などの明度を表す

この方式は次のような絵の具を使っての色の作り方と似ています。

①基本となる色をパレットに置く (H)
②白色を加えて明るくする (S)
③黒色を加えて暗くする (V)

RGB 方式の特徴は機械(モニタなど)が色を表現するのに便利な方式であるのに対し、HSV 方式は人間の感覚をもとにしているので、人間が色を作成するのに優れています。

操作方法

■起動と終了

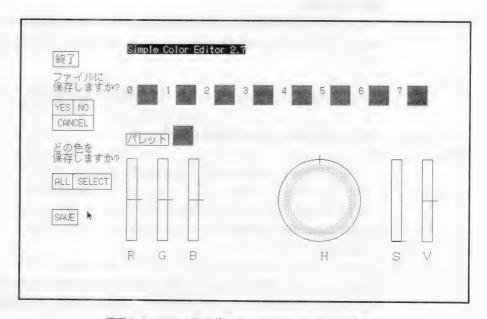
CEDIT を起動するには、ツリーエディタから"!"コマンドを使って、

cedit　－⏎

と入力します。また、MS-DOS のコマンドラインから次のように実行して結果をファイルに出力できます。

cedit　〈ファイル名〉⏎

終了するときは、画面左上の「終了」をクリックします。そうすると、「ファイルに保存しますか？　YES/NO/CANCEL」と聞いてくるので、いずれかをクリックします。

すべての色を保存するときは、「YES」をクリックした後「どの色を保存しますか？」と聞いてくるので「ALL」を選択します。

画面 9-2　CEDIT 画面(「終了」→「YES」→「SELECT」を選択)

いくつかの色だけを保存するときは「どの色を保存しますか？」と聞いてきたとき「SELECT」をクリックし、「0」〜「7」のうち保存したいアイコンだけをクリックして反転表示させます。保存する色が決まったら「SAVE」を選択します。

保存しないで終了するときは「NO」をクリックします。

また、終了しないで編集作業を続けたいときは「CANCEL」をクリックします。

■色を作成

CEDIT を起動すると、左側に RGB、右側に HSV のスライドスイッチ（アナログ的に数値を入力する画面上のスイッチ）が表示され、上方に作った色を記録する領域が 8 つ表示されます。長方形のスライドスイッチは、スイッチ内に表示されている針（横線）の位置で値を示しており、上に行くほど大きな値を示します。円形のスライドスイッチは、針の位置で色相を示しています。

スライドスイッチの使い方は、針の上でマウスをドラッグして動かします。動かすごとにスライドスイッチの示す色がパレットで変化します。また、RGB または HSV のスライドスイッチを動かすと、同じ色を示すようにもう片方のスイッチが変化します。

■色の記録

画面上に 0 から 7 までの数字のついた領域があります。ここを使って 8 つまで、作った色を記録できます。記録の方法は、マウスカーソルを数字の上に移動させて 2 回クリックします。

また、記録した色の編集も可能です。方法は、領域の番号を 1 回クリックし、次に「パレット」をクリックします。そうすると「パレット」が指定した領域と同じ色になり、スライドスイッチが該当する値を示します。

■出力

CEDIT は次の形式で結果を出力します。

```
l n, r, g, b
```

ここで"l"は mode のコマンドデータです。n は光源番号で、領域に付けられた番号に対応します。r、g、b はそれぞれの光源の RGB 成分の明るさを 0 から 1 の範囲で表したものです。

例　l 0, 1.000000, 1.000000, 1.000000　… 背景色に白を指定

10章
シェーディング実践

ここでは8章で作成した木とロボットに色を付けてみます。

10.1　木の作成　-Step 2-

まず木に色を付けてみましょう。幹と葉、背景の3つに分け、それぞれ灰色と緑色、そして水色に指定します。

■色の指定

まず解像度と視野角度を次のように指定します。

画面 10-1　解像度と視野角度の指定

背景色は水色、光源は次のように指定します。

画面 10-2　背景色と光源の指定

```
tree.mod ┬──x 213,40,0,1;
         ├──y 133,25,0,1;
         ├──i skyblue.clr  ……………… 水色の背景色ファイルの読み込み
         ├──b 0,(0,0,100)  ……………… 光源の向きを指定
         ├──b 1,(10,20,80) ……………… 光源を平行光源に指定
         ├────┬──sx 5
         │    ├──sy 5
         │    ├──sz 100
         │    └──i cone.mod
         └──i branch.mod
```

幹は灰色、葉は緑色に指定します。

画面 10-3　物体色の指定

```
tree.mod ┬──x 213,40,0,1;
         ├──y 133,25,0,1;
         ├──i skyblue.clr
         ├──b 0,(0,0,100)
         ├──b 1,(10,20,80)
         ├────┬──c 38 …………… 幹を灰色に指定
         │    ├──sx 5
         │    ├──sy 5
         │    ├──sz 100
         │    └──i cone.mod
         └────┬──c 31 …………… 葉を緑色に指定
              └──i branch.mod
```

■カラー表示

それでは完成例をカラー表示してみましょう。

①「実行用システムディスク」をドライブAに入れる。以下フロッピーディスクのみのシステム構成で説明します。
②「第2編　8章　モデリング実践」で作成したディスクをドライブBに入れる。
③ [RESET] ボタンを押す。
④次のように入力。

B>mode　tree3 [↵]

「Mode」メニューの「画像表示」を選択する（または [NFER] + [G]）と、「recalc all animation (y/n)?」と聞いてきます。[Y] [↵] を押すと、しばらくしてから画面にカラーの画像が表示されます（口絵㉒参照）。

※カラー画像が表示されるまでにはPC-9801VXで約10分かかります。
※「画像表示」を終了するときは [Q] を押します。画面上に残っているカラー画像を消去するときはCLS3コマンドを使います。
※カラー画像を表示せずにエラーメッセージを表示したままになることがあります。たとえば、同じディレクトリに必要なファイルが足りないと「File open error」、解像度と視野角度の指定がないと「resolusion not specified」というエラーメッセージが表示されます。いずれの場合も [Q] で終了してファイル内容を確認してください。
※ tree3.modの「画像表示」するには約300Kバイトの空き容量（ハードディスクで2Kバイトクラスタの場合）が必要です。

10.2 ロボットの作成　–Step 2–

次にロボットに色を付けてみましょう。背景色は白、胴体は赤と黄色、頭は青と黄色、腕と足は青、黄、緑で配色します。

■色の指定

まず、解像度や視野角度、光源、背景色などを次のように指定します。

画面 10-4　解像度、視野角度、光源と背景色の指定

```
robot.mod─┬─x 80,40,0,1;
          ├─y 50,25,0,1;
          ├─l 0,1,1,1 …………… 背景色を白に指定
          ├─l 1,1,1,1
          ├─b 0,(0,0,0)
          ├─b 1,(0,5,-1)
          ├─┬─# 左足
          │ ├─tx 3
          │ ├─tz 20
          │ └─i foot-l.mod
          ├─┬─# 右足
          │ ├─tx -3
          │ ├─tz 20
          │ └─i foot-r.mod
          ├─┬─# 身体
          │ ├─tz 20
          │ └─i body.mod
          ├─┬─# 頭
          │ ├─tz 35
```

続いて各部分の色を次のように指定します。

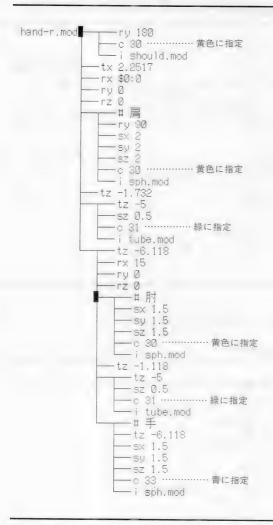
①「body.mod」を選択して胴体の色を指定。

画面 10-5　body.mod の内容

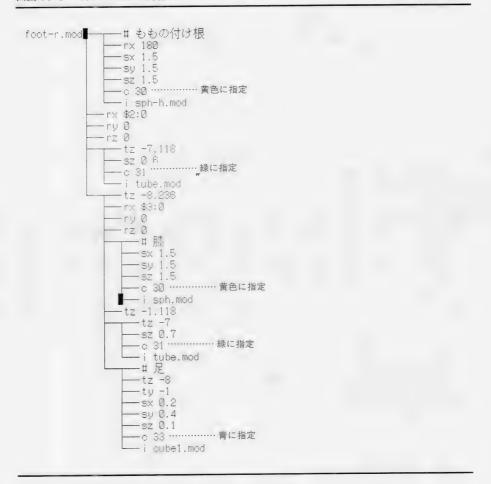
②「head.mod」を選択して首と頭の色を指定。

画面 10-6　head.mod の内容

③「hand－r.mod」を選択して右手の色を指定。
「hand－l.mod」も同様に指定します。

画面 10-7　hand－r.mod の内容

④「foot－r.mod」を選択して右足の色を指定。

「foot－r.mod」も同様に指定します。

画面 10-8　foot－r.mod の内容

■カラー表示

それでは完成例をカラー表示してみましょう。

①「実行用システムディスク」をドライブAに入れる。以下フロッピーディスクのみのシステム構成で説明します。
②「第2編　8章　モデリング実践」で作成したディスクをドライブBに入れる。
③ RESET ボタンを押す。
④次のように入力。

B>mode　robot2 ↵

「Mode」メニューから「画像連続表示」を選択する(または NFER ＋ A)と、「recalc　all animation (y/n)?」と聞いてきます。 Y ↵ を押すと、しばらくしてから画面にカラー画像が連続して表示されます(口絵㉓〜㉛参照)。

※カラー画像が表示されるまでには1画面あたりPC-9801VXで約10分、合計約90分かかります。
※「画像連続表示」を終了するときは Q を押します。画面上に残っているカラー画像を消去するときはCLS3コマンドを使います。
※カラー画像を表示せずにエラーメッセージを表示したままになることがあります。たとえば、同じディレクトリに必要なファイルが足りないと「File open error」、解像度と視野角度の指定がないと「resolusion not specified」というエラーメッセージが表示されます。いずれの場合も Q で終了してファイル内容を確認してください。
※ robot2.modの「画像連続表示」するには約1160Kバイトの空き容量(ハードディスクで2Kバイトクラスタの場合)が必要です。なるべくハードディスクで使用してください。

Appendix

ここではテキストエディタを使った応用例や
付属のソースファイルのコンパイル方法を説明し、
パッケージにある全実行ファイルを一覧にしています。

Appendix A
テキストエディタを使ったデータ編集

モデルデータはテキストファイル形式で保存されますので、テキストエディタを使って編集できます。すでに座標の決まったポリゴンを入力する場合や簡単な引数の変更を行う場合などに利用するとよいでしょう。

記述が間違っているとMODEで読み込めないので注意してください。

使用できる文字

文　字	名　前	使用法
012−9	数　字	数値やファイル名に使用
ABC−Z	大文字	ファイル名に使用
abc−z	小文字	コマンドやファイル名に使用
,	カンマ	引数列の区切りに使用
.	ピリオド	小数点を表す
;	セミコロン	不定個数の引数列を要求するコマンドで終了を表す
−	負符号	負の数値を表す
+	正符号	正の数値を表す
(	左カッコ	座標を表す
)	右カッコ	座標を表す
{	左中カッコ	コマンドの階層化に使用
}	右中カッコ	コマンドの階層化に使用
#	イゲタ	コメントに使用
$	ドルマーク	変数を表す

表 A-1　使用できる文字一覧

モデルデータ内で使用できる文字は、**表 A-1** およびスペース、タブ、改行です。英字の大文字と小文字は区別しますので、注意してください。

これらの文字の並びからトークンが構成されます。トークンには、シンボル、数値、カンマ、セミコロン、カッコがあります。

シンボルは、コマンド名やファイル名を表すのに使用します。数値は、コマンドへの引数として使用し、符号(+、−)、数字、ピリオド(小数点)で表します。変数は、〝$〟と数字で表します(たとえば$16)。カンマは、コマンドの引数列の区切りに使用します。セミコロンは、不定個数の引数列を要求するコマンドで終了を表すのに使用します。カッコは、X、Y、Zからなる座標を表すために使用します。中カッコは、階層化コマンドとして使用します。

スペースやタブなどからなる空白は、トークン間の分離に使用します。空白を挿入できるのは、行の先頭と終端、そして続くコマンドの書式の説明の中で、スペースの存在している位置です。〝#〟から行末までは、コメントとして解釈されます。

コマンドの書式

テキストエディタとツリーエディタとの表示上の違いは階層化コマンド({})と形状モデル用コマンド(p)です。MODEのツリーエディタではポリゴンの数値を直接入力できませんが、テキストエディタではポリゴンの数値を直接変更できます。

モデルファイル中の1行は、2文字以内のコマンド名とそれに続く引数から構成されますが、空白や区切り文字などの位置で途中で改行して2行に分けても構いません。また、引数の個数が不定の場合は、終端をセミコロンで示します。

■形状モデル用コマンド

MODEのポリゴンエディタに相当するのが形状モデル用コマンドです。次のように記述します。

書　式

```
p(x, y, z), …, (x, y, z);
p(x, y, z, nx, ny, nz), …, (x, y, z, nx, ny, nz);
```

例

```
p(1, 1, 1), (1, 1, -1), (1, -1, -1), (1, -1, 1);
```

コマンド名は〝p〟です。引数は、ポリゴンを構成する点の座標をカンマで区切ってすべて記述し、最後にセミコロンをおきます。このコマンドで指定されるポリゴンは、座標を指定された順番に結んだものとなります。座標は最低3つ指定する必要があります。

「nx, ny, nz」の引数は、各点における法線ベクトルを指定します。法線ベクトルとは、レンダリングの際に使用する値で、各点の座標の後に続けて記述します。

■階層化コマンド

書　式

```
{
}
```

引　数　無し

階層化コマンドは引数を持たず、〝｛〞と〝｝〞で対にして使用し、〝｛〞と〝｝〞の間にあるデータをブロックとして指定します。このコマンドによる階層化は何重になっても構いません。

エディタで入力する際、階層化コマンドで囲まれた行は、次のように先頭にタブを入れるとデータの位置するブロックが分かりやすくなります。また、i コマンドで読み込むモデルファイルは、読み込む側のデータに影響を及ぼさないように、全体を階層化コマンドで囲む必要があります。

例

```
{
  tx10
  {
    ry45
    i   sphere.mod
  }
  i   cube.mod
}
```

Appendix B ソースファイルのコンパイル

添付のディスクには本書で紹介したプログラムのソースファイルを掲載しています。ここではそれぞれのプログラムのコンパイル方法を示します。C言語に関するコンパイルソースのテストは米国Microsoft社のMicrosoft C Professional Development System Version 6.0（以下MS-Cと記述）、アセンブルについては同Microsoft Macro Assembler Version 5.10（以下MASMと記述）を使って行います。

■用意するもの

- Microsoft C Professional Development System Version 6.0
- Microsoft Macro Assembler Version 5.10
- 『MS-DOS SOFTWARE TOOLS 1』（弊社刊）のRMコマンド（MODEコマンドをMAKEFILEを使ってコンパイルするときに必要です）
- 初期化済みのディスク（ハードディスクを使う場合は、空き容量を確認してください）

■コンパイル手順

「保存用ソースディスク」には各コマンドごとにディレクトリがあります。

ディレクトリ	内　容
¥CEDIT	CEDIT コマンド
¥CONE	CONE コマンド
¥CUBE	CUBE コマンド
¥DITHER	DITHER コマンド
¥DRAW	DRAW コマンド
¥CLS3	CLS3 コマンド
¥GLOAD	GLOAD コマンド
¥GSAVE	GSAVE コマンド
¥LGBIOS	グラフィックライブラリ〝LGBIOS″
¥MODE	MODE コマンド
¥MODE¥BITMAPS	MODE コマンド(アイコンデータ)
¥MODEL	MODEL コマンド
¥MOTION	MOTION コマンド
¥PLANE	PLANE コマンド
¥POLHD	POLHD コマンド
¥RENDER	RENDER コマンド
¥ROT	ROT コマンド
¥SPHERE	SPHERE コマンド
¥THICK	THICK コマンド
¥TUBE	TUBE コマンド

表 B-1 「保存用ソースディスク」のディレクトリ

以下の説明では、ハードディスクからMS-DOSを立ち上げ(ドライブA)、「保存用ソースディスク」をドライブB、コピーしたいファイルをコピーするディスクのドライブをドライブCとして説明します。

① MS-CおよびMASMのあるディレクトリにパスが通っているかどうか確認する。
②「保存用ソースディスク」からコンパイルしたいファイルを選んで別のディスクにコピーする。

MODEコマンド

MODE コマンドは¥MODE、¥MODE¥BITMAPS ディレクトリの中にあるファイルすべてから作成します。

●ファイルの内容

MODE コマンドのソースファイルは、¥MODE、¥MODE¥BITMAPS にあります。ファイルの内容は次のとおりです。

ファイル名	内　容
MAKEFILE	メイクファイル
BUFFER.C	それぞれのエディタの画面情報を管理
COMMANDS.C	ポリゴンエディタのそれぞれの機能を実現
DATA.H	変数、定数などを定義
EXTERN.H	様々な関数を宣言
FILEIO.C	ファイル入出力用
GEDIT.C	ポリゴンエディタを実現
GEDIT.H	ポリゴンエディタで利用している変数、定数などを定義
GRAPHICS.ASM	グラフィックスを描画
GRAPHICS.OBJ	グラフィックライブラリ〝GRAPHICS〞
GSELECT.C	ポリゴンエディタで点やポリゴンを選択
GUNDO.C	ポリゴンエディタでアンドゥを実現
IO.C	ノードエディタでノードを編集
ITEM.C	ポリゴンエディタでメッシュなどを表示
MAIN.C	モードの起動
MATRIX.C	行列計算用
MENU.C	メニューの表示、選択用
MOTION.C	モーションエディタ
SCREEN.C	画面上で文字列を編集
STRUCT.C	ポリゴンなどの構造体を変更するための関数を収録
STRUCT.H	ポリゴンなどの構造体を定義
SYSTEM.C	システムに依存した部分を収録
SYSTEM.H	システムに依存した部分について定義
TREE.C	ツリーエディタ
WINDOW.C	ポリゴンエディタのウィンドウを描画

表 B-2　¥MODE の内容

ファイル名	内　容	ファイル名	内　容
ATNORM.ICN	"法線生成" アイコン	NRMVED.ICN	"法線変更" アイコン
AXIS.ICN	"座標軸" アイコン	PERS.ICN	"透視図" アイコン
CENTER.ICN	"センタリング" アイコン	POINT.ICN	"選択点指定モード" アイコン
CLR.ICN	"選択点解除" アイコン	REDUCT.ICN	"スケール変更" アイコン
COPY.ICN	"複写" アイコン	RESET.ICN	"リセット" アイコン
CSR.ICN	"カーソル移動モード" アイコン	ROTATE.ICN	"回転" アイコン
DELETE.ICN	"削除" アイコン	ROTX.ICN	"視点回転Ｘ軸" アイコン
FOUR.ICN	"三面図と透視図" アイコン	ROTY.ICN	"視点回転Ｙ軸" アイコン
FRONT.ICN	"前面図" アイコン	ROTZ.ICN	"視点回転Ｚ軸" アイコン
HELP.ICN	"ヘルプ" アイコン	SCALE.ICN	"拡大" アイコン
INSERT.ICN	"中点作成" アイコン	SCALEDWN.ICN	"カーソル縮小" アイコン
MAKEPOL.ICN	"ポリゴン作成" アイコン	SCALEUP.ICN	"カーソル拡大" アイコン
MESH.ICN	"メッシュ" アイコン	SETPOINT.ICN	"点作成" アイコン
MOVE.ICN	"移動" アイコン	SIDE.ICN	"側面図" アイコン
MOVEXY.ICN	"上下左右移動" アイコン	SUIT.ICN	"フィット" アイコン
MOVEZ.ICN	"前後移動" アイコン	TOP.ICN	"上面図" アイコン
NRMV.ICN	"法線表示／非表示" アイコン	UNDO.ICN	"アンドゥ" アイコン

表 B-3　¥MODE¥BITMAPS の内容

●コンパイル方法

①¥MODE、¥MODE¥BITMAPS ディレクトリの内容をすべて初期化済みのディスクにコピー。

```
A>c：⏎
C>md  bitmaps⏎
C>copy  b：¥mode  c：⏎
C>copy  b：¥mode¥bitmaps  c：bitmaps⏎
```

②次のように入力(MODE.EXE は省略可能)。

```
C>nmake  mode.exe⏎
```

※ NMAKE は MS-C に付属のコマンドです。
※ MAKEFILE の内容はコマンドごとに違います。複数のコマンドをコンパイルするときは、かならず別のディスクか別のディレクトリにコピーしてから行ってください。

そうすると、ドライブ C のルートディレクトリに「mode.exe」が作成されます。

CEDITコマンド

ファイル名	内　容	ファイル名	内　容
MAKEFILE	メイクファイル	HSV2RGB.C	HSV2RGB ソースファイル
CEDIT.C	CEDIT ソースファイル	MSG.H	MSG ヘッダファイル
CEDIT.H	CEDIT ヘッダファイル	PLOT.C	PLOT ソースファイル
DATA.H	DATA ヘッダファイル	SINCOS.C	SINCOS ソースファイル
DITHER.C	DITHER ソースファイル	STRUCT.H	STRUCT ヘッダファイル
EXTERN.H	EXTERN ヘッダファイル	SYSTEM.C	SYSTEM ソースファイル
GRAPHICS.ASM	GRAHICS ソースファイル	SYSTEM.H	SYSTEM ヘッダファイル

①¥CEDIT ディレクトリの内容をすべて別のディスクにコピー。
② nmake　cedit.exe ⏎ と入力。

PLANEコマンド

ファイル名	内　容	ファイル名	内　容
MAKEFILE	メイクファイル	GRAPHICS.ASM	GRAPHICS ソースファイル
MOUSE.C	マウス／画面入出力ルーチン	GRAPHICS.H	GRAPHICS ヘッダファイル
PLANE.C	PLANE ソースファイル	GRAPHICS.OBJ	グラフィックライブラリ〝GRAPHICS〟

①¥PLANE ディレクトリの内容をすべて別のディスクにコピー。
② nmake　plane.exe ⏎ と入力。

DRAWコマンド

ファイル名	内　容
MAKEFILE	メイクファイル
DRAW.C	DRAW ソースファイル

①¥DRAW ディレクトリの内容をすべて別のディスクにコピー。
② nmake　draw.exe ⏎ と入力。

CLS3コマンド

ファイル名	内　容
MAKEFILE	メイクファイル
CLS3.C	CLS3 ソースファイル
LGBIOS.OBJ	グラフィックライブラリ〝LGBIOS〞

CLS3.C は現在画面上に表示されているグラフィック画面を消去するプログラムです。このプログラムでは、〝LGBIOS.OBJ〞を使います。

①¥CLS3 ディレクトリの内容をすべて別のディスクにコピー。
② nmake　cls3.exe ⏎ と入力。

CUBEコマンド

ファイル名	内　容
MAKEFILE	メイクファイル
CUBE.C	CUBE ソースファイル

①¥CUBE ディレクトリの内容をすべて別のディスクにコピー。
② nmake　cube.exe ⏎ と入力。

CONEコマンド

ファイル名	内　容
MAKEFILE	メイクファイル
CONE.C	CONE ソースファイル

①¥CONE ディレクトリの内容をすべて別のディスクにコピー。
② nmake　cone.exe ⏎ と入力。

POLHDコマンド

ファイル名	内　容
MAKEFILE POLHD.C	メイクファイル POLHD ソースファイル

①¥POLHD ディレクトリの内容をすべて別のディスクにコピー。
② nmake　polhd.exe ⏎ と入力。

TUBEコマンド

ファイル名	内　容
MAKEFILE TUBE.C	メイクファイル TUBE ソースファイル

①¥TUBE ディレクトリの内容をすべて別のディスクにコピー。
② nmake　tube.exe ⏎ と入力。

ROTコマンド

ファイル名	内　容
MAKEFILE ROT.C	メイクファイル ROT ソースファイル

①¥ROT ディレクトリの内容をすべて別のディスクにコピー。
② nmake　rot.exe ⏎ と入力。

THICKコマンド

ファイル名	内　容
MAKEFILE	メイクファイル
THICK.C	THICK ソースファイル

①¥THICK ディレクトリの内容をすべて別のディスクにコピー。
② nmake　thick.exe [↵] と入力。

MODELコマンド

ファイル名	内　容
MAKEFILE	メイクファイル
MODEL.C	MODEL ソースファイル
LGBIOS.OBJ	グラフィックライブラリ “LGBIOS”

MODEL.C はモデリング・サブシステムのプログラムです。このプログラムでは、ワイヤフレーム表示のために“LGBIOS.OBJ”というアセンブラのルーチンを使います。

①¥MODEL ディレクトリの内容をすべて別のディスクにコピー。
② nmake　model.exe [↵] と入力。

SPHEREコマンド

ファイル名	内　容
MAKEFILE	メイクファイル
SPHERE.C	SPHERE ソースファイル

SPHERE.C は球面を近似するポリゴンを出力するプログラムです。コンパイルは次のように行います。

①¥SPHERE ディレクトリの内容をすべて別のディスクにコピー。
② nmake　sphere.exe [↵] と入力。

MOTIONコマンド

ファイル名	内　容
MAKEFILE	メイクファイル
MOTION.C	MOTION ソースファイル

MOTION.C はモーションデザイン・サブシステムです。コンパイルは次のように行います。

①¥MOTION ディレクトリの内容をすべて別のディスクにコピー。
② nmake　motion.exe [⏎] と入力。

RENDERコマンド

ファイル名	内　容
MAKEFILE	メイクファイル
RENDER.C	RENDER ソースファイル

RENDER.C はレンダリング・サブシステムです。通常は次のようにしてコンパイルします。

①¥RENDER ディレクトリの内容をすべて別のディスクにコピー。
② nmake　render.exe [⏎] と入力。

DITHERコマンド

ファイル名	内　容
MAKEFILE	メイクファイル
DITHER.C	DITHER ソースファイル
LGBIOS.OBJ	グラフィックライブラリ〝LGBIOS〃

DITHER.Cはタイリングとディザリングを使って画面に画像を出力するプログラムです。このプログラムにも〝LGBIOS.OBJ〃が必要で、コンパイルは次のように行います。

①¥DITHERディレクトリの内容をすべて別のディスクにコピー。
② nmake　dither.exe ⏎ と入力。

GSAVEコマンド

ファイル名	内　容
GSAVE.ASM	GSAVE ソースファイル

GSAVE.ASMは、PC−9800シリーズ(XAを除く)の画面(グラフィックメモリ)のデータをファイルとして保管するためのプログラムです。アセンブルは次のようにして行います。
①¥GSAVEディレクトリの内容をすべて別のディスクにコピー。
②次のように入力して〝GSAVE.OBJ〃を作成。

```
C>masm　gsave ; ⏎
```

③〝LINK〃コマンドを使って〝GSAVE.EXE〃を作成。

```
C>link　gsave.obj,　gsave.exe,　NUL ; ⏎
```

GLOADコマンド

ファイル名	内　容
GLOAD.ASM	GLOAD ソースファイル

GLOAD.ASM はGSAVE コマンドで作成したファイルを読み込んで画面に表示するためのプログラムです。

アセンブルとリンクの方法は、次のようになります。

①¥GLOAD ディレクトリの内容をすべて別のディスクにコピー。
②次のように入力。

```
C>masm  gload ; ⏎ .
C>link  gload.obj,   gload.exe,   NUL ; ⏎
```

グラフィックライブラリ "LGBIOS"

ファイル名	内　容
LGBIOS.ASM	LGBIOS ソースファイル

〝LGBIOS.ASM〟はPC－9800シリーズ(XAを除く)用のグラフィックライブラリです。このプログラムは次のようにしてアセンブルします。

①¥LGBIOS ディレクトリの内容をすべて別のディスクにコピー。
②次のように入力。

```
C>masm  /MX  lgbios ; ⏎
```

アセンブルすると〝LGBIOS.OBJ〟が作成されます。ここで〝/MX〟オプションを忘れるとMODEL コマンド、DITHER コマンドをコンパイルする際にリンクできませんので注意してください。

Appendix C
クイックリファレンス
——MODEシステム——

C.1　MODEの起動

書　式

MODE　[－m　〈モーションファイル〉　〈モデルファイル〉]　[〈モーションファイルとモデルファイルのベースネーム〉]

機　能

MODEコマンドは、〈モーションファイルとモデルファイルのベースネーム〉で指定されたベースネームでモーションファイル(拡張子.mot)とモデルファイル(拡張子.mod)を作成します。

オプション

－m　　モデルファイルと違うベースネームのモーションファイルを読み込む。

解　説

MODEコマンドは、モデリングとモーションデザインを行うエディタです。〈モデルファイル〉のみ指定した場合はモデルファイルだけを読み込みます。モーションファイルだけを読み込みたい場合は「mode　－m　〈モーションファイル〉」とします。

実行例

mode　－m　test.mot　sample.mod
mode　sample

C.2 MODEメニュー

```
ツリーエディタ        NFER+T
ポリゴンエディタ      NFER+P
モーションエディタ    NFER+M
ファイル              NFER+F
シェル                NFER+S
------------------------------
画像表示              NFER+G
画像連続表示          NFER+A
終了                  NFER+Q
```

画面 C-1　MODE メニュー画面

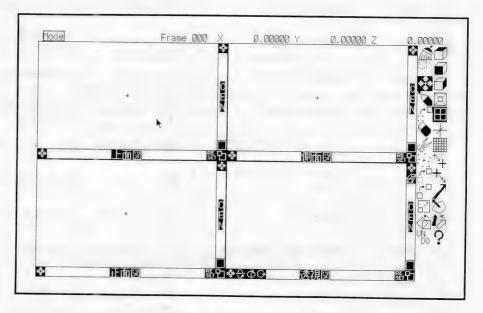
メニュー間の移動をキー操作で行う場合は次のようにします。このキー操作は「ツリーエディタ」、「ポリゴンエディタ」、「モーションエディタ」、「ファイルの切り替え」に共通です。

機　能	キー操作
ツリーエディタへ	NFER + T (XFER + T)
ポリゴンエディタへ	NFER + P (XFER + P)
モーションエディタへ	NFER + M (XFER + M)
ファイルの切り替え	NFER + F (XFER + F)
シェルの起動	NFER + S (XFER + S)
画像表示	NFER + G (XFER + G)
画像連続表示	NFER + A (XFER + A)
MODE の終了	NFER + Q (XFER + Q)

C.3　ポリゴンエディタ

■概　要

ポリゴンエディタでは、ポリゴンの作成、編集、それに動画のチェックができます。編集作業は、画面右に配置されたアイコンをマウスカーソルでクリックして実行します。

ポリゴンは、カーソル移動モードで点を作り、選択点指定モードで任意の点を選択点として作成します。編集するときは、選択点指定モードで任意の点を選択点とし、編集内容を表すアイコンをクリックします。編集は常に点に対して行われ、ポリゴンには作用しません。

画面 C-2　ポリゴンエディタ画面

■動作モード

- ●選択点指定モード　　選択点を選択するモード
- ●カーソル移動モード　　エディットカーソルを移動させるモード

■アイコン

アイコン	機　能	アイコン	機　能
(選択点解除)	すべての点を選択されていない状態にする	(カーソル縮小)	エディットカーソルを縮小
(選択点指定モード)	選択点指定モードにする	(センタリング)	エディットカーソルをすべての選択点の中心へ移動
(カーソル移動モード)	カーソル移動モードにする	(移動)	選択点または選択ブロックを移動
(三面図と透視図)	三面図および透視図を表示(初期状態)	(複写)	選択点または選択ブロックを複写
(透視図)	透視図のみを表示	(回転)	選択点または選択ブロックを回転
(正面図)	正面図のみを表示	(拡大/縮小)	エディットカーソルを中心にして選択点を拡大または縮小
(側面図)	側面図のみを表示	UN DO (アンドゥ)	直前の処理の取り消し
(上面図)	上面図のみを表示	(ヘルプ)	クリックした場所のアイコンなどの説明を表示
(点作成)	エディットカーソルの座標に点を作成	(上下左右移動)	図を上下左右に移動
(ポリゴン作成)	選択点をつなぎ、1つのポリゴンを作成	RE SET (リセット)	画面の表示状態を初期化
(削除)	選択点を削除	(スケール変更)	透視図の視点を移動
(座標軸)	座標軸の表示、非表示の選択	(フィット)	物体を画面に合わせて表示
(中点作成)	選択点の間を等分した座標に点を作成	C E N (センター)	エディットカーソルが画面の中心になるように表示
(メッシュ)	三面図にメッシュをかける	(X軸視点回転)	透視図の視点をX軸を中心にして回転
(カーソル拡大)	エディットカーソルを拡大	(Y軸視点回転)	透視図の視点をY軸を中心にして回転
(法線生成)	法線ベクトルを生成	(Z軸視点回転)	透視図の視点をZ軸を中心にして回転
(法線表示/非表示)	法線ベクトルの表示、非表示の切り替え	(前後移動)	図を前後に移動
(法線変更)	法線ベクトルの変更を開始		

■キー操作

キー操作	内　容
N	次のコマを表示
スペース	次のコマを表示
P	前のコマを表示
CTRL + H	前のコマを表示
CTRL + L	画面の再描画
SHIFT + ?	キー操作の説明を表示

C.4 ツリーエディタ

■概　要

ツリーエディタで行う処理は、コマンドデータの編集とブロックの操作、そしてツールの呼び出しです。ツリーエディタではすべての操作をキーボードから行います。

コマンドデータの編集とは、ポリゴン以外のデータ(r、t、s、x、y、c、b、l、i、#)の挿入や削除などの作業です。ブロックの操作とは、階層化コマンドを使用してデータを階層化することで、階層化の状態を視覚的に分かりやすく表示します。ツールの呼び出しとは、特定の図形データを作るためのプログラムを呼び出す作業です。

■キー操作

●カーソル移動

キー操作	内　容
CTRL + B	1画面上に移動
CTRL + U	半画面上に移動
CTRL + F	1画面下に移動
CTRL + D	半画面下に移動
CTRL + R	カーソルが画面の中心になるように表示
H	1つ高いレベル(左)へ移動
L	1つ低いレベル(右)へ移動
J	1つ下へ移動
K	1つ上へ移動
SHIFT + J	1つ下のブロックへ移動
SHIFT + K	1つ上のブロックへ移動
M	カーソル位置に印を付ける
SHIFT + ’	印を付けたデータに移動

● データ編集

キー操作	内　容
O	ダミーのデータをカーソル位置の後ろに挿入
SHIFT + O	ダミーのデータをカーソル位置の前に挿入
P	バッファのデータをカーソル位置の後ろに挿入
SHIFT + P	バッファのデータをカーソル位置の前に挿入
D	カーソル位置のデータを削除し、バッファに入れる
Y	カーソル位置のデータをバッファに入れる
SHIFT + >	カーソル位置のデータのレベルを１つ下げる
SHIFT + <	カーソル位置のデータのレベルを１つ上げる
C	データの引数を変更
SHIFT + C	データのタイプを変更
SHIFT + $	データの引数が定数のときは変数に、変数のときは定数に変更

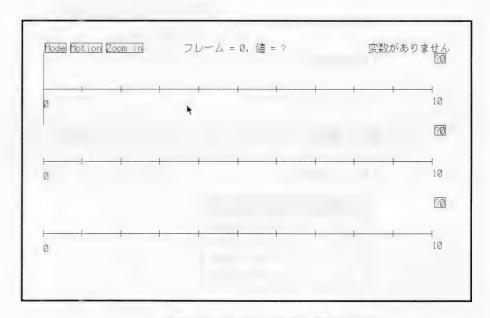
● その他の操作

キー操作	内　容
スペース	ポリゴンデータの座標を表示。またはｉコマンドで読み込み指定したファイルの編集
CTRL + L	画面の再描画
U	直前の動作の取り消し
E	他のファイルの編集
W	データを画面上のファイル名で保存
R	ファイル名の変更
SHIFT + !	ツールの使用
SHIFT + ?	説明を表示
数字キー	繰り返し処理。操作の前に数字キーで繰り返したい回数を入力

C.5 モーションエディタ

■概　要

モーションエディタは、ツリーエディタで書き込んだ変数に値を与えるパラメータファイルを作るためのエディタです。

基本的な操作は、横に伸びる時間軸の上にキーフレームをおき、その時間での変数の値を指定します。使用を宣言した変数の数だけ指定します。

画面 C-3　モーションエディタ画面

■メニューの操作

● Motion

新しいフレームを作る	カーソル位置に点を作成
フレームを消す	カーソル位置の点を削除

● Zoom in

時間軸の縮尺を変更します。

●変数(画面右のメニュー)

変数の使用を宣言します。

■キー操作

●カーソル移動

キー操作	内　容
H	カーソルを左に移動
J	カーソルを下に移動
K	カーソルを上に移動
L	カーソルを右に移動

●画面の操作

キー操作	内　容
SHIFT+H	時間軸の表示範囲を左に移動
SHIFT+L	時間軸の表示範囲を右に移動
CTRL+L	画面の再描画

● 変数の変更

キー操作	内　容
SHIFT + J	次の変数を表示
SHIFT + K	前の変数を表示

● 縮尺の変更

キー操作	内　容
SHIFT + *	時間軸の間隔を拡大
/	時間軸の間隔を縮小

● その他の操作

キー操作	内　容
W	モーションファイルとパラメータファイルにデータを書き込む
SHIFT + ?	説明を表示

C.6 ファイルの切り替え

■概　要

MODE は複数のデータファイルをメモリに読み込んで編集できます。しかし、ポリゴンエディタやツリーエディタでは同時に2つ以上のデータファイルを編集できません。そこで、どのファイルを編集するかを選択します。

画面 C-4　ファイルの切り替え画面

■カーソル移動

キー操作	内　容
CTRL + B	1画面上に移動
CTRL + U	半画面上に移動
CTRL + F	1画面下に移動
CTRL + D	半画面下に移動
H	左へ移動
L	右へ移動
J	1つ下へ移動
K	1つ上へ移動

■その他の操作

キー操作	内　容
スペース	カーソル位置のファイルを編集
CTRL + L	画面を再描画
R	ファイル名を変更
SHIFT + ?	説明の表示

Appendix D
コマンドリファレンス
――ツリーエディタで使えるツール――

ポリゴンは、ポリゴンエディタで作る以外に、ツリーエディタ上でツールを使って作ることもできます。ツールを使用すれば、幾何学的な物体でも簡単に作れます。

CEDIT　色情報の生成

書　式

CEDIT　[ー]　[〈ファイル名〉]

機　能

CEDIT コマンドは、ツリーエディタの〝l″コマンドで使用する色情報を生成します。〝ー″を指定すると結果を標準出力に出力し、〈ファイル名〉を指定すると結果をファイルに出力します。

オプション

ー　　　結果を標準出力に出力する。

解　説

CEDIT コマンドは、〝1　0″～〝1　7″までの色情報を生成するツールです。〈ファイル名〉を指定するとき拡張子(.clr)は省略できます。

実行例

```
cedit　sample
cedit　ー
```

PLANE

選択点の順番でポリゴンを生成

書　式　PLANE ［－］［<ファイル名>］

機　能　PLANEコマンドは、選択点の順番でポリゴンを生成します。"－"を指定すると結果を標準出力に出力し、<ファイル名>を指定すると結果をファイルに出力します。

オプション　－　　結果を標準出力に出力する。

解　説　ポリゴンエディタを使ってポリゴンを作ると、選択点を結ぶ順番が自動的に決められてしまうため、凹部のあるポリゴンを作るときなどは多少手間がかかります。PLANEコマンドでは選択点を作った順にポリゴンが作成できます。<ファイル名>を指定するとき拡張子(.mod)は省略できます。PLANEコマンドでは既存のファイルの編集はできません。

実行例

```
plane  sample
plane  －
```

CONE

円錐のポリゴンデータを生成

書　式　CONE <分割数>

機　能　CONEコマンドは、<分割数>で指定された面の数で円錐の側面を近似して、円錐のポリゴンデータを生成します。

解　説　CONEコマンドは、円錐のデータを作るツールです。<分割数>はいくつの面で円を近似するかを表し、自然数で与えます。ポリゴンデータは、円錐の頂点が(0, 0, 1)に、底面はXY平面上に作成されます。

実行例

```
cone  10
```

CUBE　立方体の角をカットしたポリゴンデータを生成

書式　CUBE　〈辺の長さ〉〈角の長さ〉

機能　CUBE コマンドは、〈辺の長さ〉で指定された立方体の辺の長さと〈角の長さ〉で指定されたカットしてできる三角形の辺の長さにしたがって、立方体の角をカットしたポリゴンデータを生成します。

解説　CUBE コマンドは、立方体の角を削った図形のデータを出力するツールです。引数は、辺の長さと角を削る幅を実数で与えます。辺の長さは立方体の辺の長さ、角を削る幅は角を削ってできた正三角形の1辺の長さです。角を削る幅は、0から立方体の辺の長さの1/sqr(2)倍までの範囲で指定します。
このツールで得られたデータを組み合わせて文字の形などを作ると、見栄えのよいものができます。

実行例　cube　10　3

SPHERE　球のポリゴンデータを生成

書式　SPHERE　〈分割数〉

機能　SPHERE コマンドは、〈分割数〉で指定された面の数で近似した球のポリゴンデータを生成します。

解説　SPHERE コマンドは、原点を中心にした半径1の球のデータを生成します。〈分割数〉とは、いくつの面で球のデータを近似するかを表す数のことです。たとえば分割数が10ならば、全部で100(＝10×10)の面で球のデータが作られます。分割数が大きいほど球に近いデータが得られますが、データが大きくなってポリゴンエディタでの表示が遅くなります。
中心が原点で半径1というのは固定なので、必要に応じて〝t〟コマンドで移動させ、〝s〟コマンドで大きさを変えて使います。

実行例　sphere　10

POLHD

正多面体のポリゴンデータを生成

書　式　POLHD　<面数>

機　能　POLHDコマンドは、<面数>で指定された値で正多面体のポリゴンデータを生成します。

解　説　POLHDは、正多面体のデータを作るツールです。面数には実際に存在する多面体の面数だけが指定可能です。つまり、4、6、8、12、20のいずれかになります。多面体の大きさは、半径1の球に収まる程度になります。

実行例　polhd　20

TUBE

円柱のポリゴンデータを生成

書　式　TUBE　[−t]　<半径>　<厚み>　[<分割数>]

機　能　TUBEコマンドは、<半径>で指定された半径の長さ、<厚み>で指定された円柱の長さ、<分割数>で指定された面数で円柱側面を近似してポリゴンデータを生成します。

オプション　−t　　底面と上面のポリゴンデータを作らない(円筒を作成)。

解　説　TUBEは、円柱のポリゴンデータを作るツールです。引数は<半径>と<厚み>を実数で、そして分割数を自然数で与えます。<半径>は底面および上面の半径の長さで、<厚み>は円柱の長さです。<分割数>はいくつの面で円を近似するかを表し、省略した場合は半径の12倍となります。
出力されるデータは、側面、上面、底面の順に作成されるので、上面や底面が必要ないときは最後の2つのポリゴンデータを削除します。

実行例　tube　1　10　10

ROT

ポリゴンデータをもとに回転体を作成

書　式　ROT　〈モデルファイル〉〈分割数〉

機　能　ROTコマンドは、〈モデルファイル〉で指定されたポリゴンデータをもとに〈分割数〉で指定された面数で回転体を生成します。

解　説　ROTコマンドは、〈モデルファイル〉で指定されたポリゴンデータ、おもに平面から〈分割数〉で指定された面数で回転体を生成します。デフォルト値は10です。

実行例　rot　glassp.mod　10

THICK

ポリゴンデータに厚みを付加

書　式　THICK　〈モデルファイル〉〈厚み〉

機　能　THICKコマンドは、〈モデルファイル〉で指定されたポリゴンデータをもとに〈厚み〉で指定された数値で厚みのついたポリゴンデータを生成します。

解　説　THICKコマンドは、〈モデルファイル〉で指定されたポリゴンデータ、おもに平面から〈厚み〉で指定された数値で厚みのついたポリゴンデータを生成します。

実行例　.thick　circle.mod　10

Appendix E
コマンドリファレンス
――コマンドラインから使うツール――

DRAW　イメージファイルを生成

書式

DRAW ［−x 〈解像度〉, 〈タイルサイズ〉 −y 〈解像度〉, 〈タイルサイズ〉］ ［−f 〈フレームA〉 −t 〈フレームZ〉］ ［−p 〈パラメータファイル・ベースネーム〉］ ［−g 〈画像ファイル・ベースネーム］ 〈モデルファイル〉

機能

DRAWコマンドは、〈モデルファイル〉と各引数から画像ファイルを生成、表示します。

オプション

−x、−y　解像度とタイルサイズの指定。パラメータのデフォルト値は、「−x 213, 3 −y 133, 3」。

−f、−t　フレームの指定。「−f」が開始するフレームで、「−t」が終了するフレーム。

−p　パラメータファイルのベースネームの指定。

−g　画像ファイルのベースネームの指定。

解説

DRAWコマンドを実行すると「recalc all animation (y/n)?」と聞いてきます。新たにデータを作成するときは［Y］［↵］、すでにあるデータを表示するときは［N］［↵］を押します。いずれの場合も画像を表示し続けるので注意してください。［Q］を押すと終了します。また、画像の消去にはCLS3コマンドを使います。

DRAWコマンドの実行にはDITHER、RENDER、GSAVE、GLOADおよびMODELコマンドが必要です。

実行例

draw −x 80, 3 −y 50, 3 −f sample.000 −t sample.010 −p pic_prm −g pic_g sample.mod

DITHER　RENDERが出力したイメージファイルを画面に表示

書　式　DITHER　[−x　〈解像度〉,〈タイルサイズ〉　−y　〈解像度〉,〈タイルサイズ〉]　〈イメージファイル〉

機　能　DITHERコマンドは、RENDERが出力したイメージファイルを画面に表示します。

オプション　−x、−y　　解像度とタイルサイズの指定。パラメータのデフォルト値は、「−x　213，3　−y　133，3」。

解　説　DITHERコマンドはMODELとRENDERによってモデルファイルから作られたイメージファイルを画面に表示します。オプションによる指定の解像度は、あらかじめモデルファイルの中で指定したものと等しくなければなりません。

実行例

```
dither　−x　320，2　−y　200，2　sample.pic
dither　−x　640，1　−y　400，1　sample.pic
```

RENDER　MODELコマンドが出力したデータからイメージファイルを生成

書　式　RENDER　〈イメージファイル〉　<　〈MODELコマンドが出力したデータ〉

機　能　RENDERコマンドは、MODELコマンドが出力したデータからイメージファイルを生成します。

解　説　RENDERは、MODELコマンドによって生成し出力したRENDER用のデータからイメージファイルを生成するツールです。MODELコマンドが出力したデータをあらかじめファイルにしておいてもよいですが、パイプでつなげて一括して処理したほうがよいでしょう。

実行例

```
render　sample.pic　<　sample.out
```

GSAVE GLOAD　画面上のイメージファイルのセーブ／ロード

書　式　GSAVE　<イメージファイル>

機　能　GSAVE コマンドは、画面に表示されているイメージファイルをセーブします。

書　式　GLOAD　<イメージファイル>

機　能　GLOAD コマンドは、GSAVE でセーブしたイメージファイルを画面にロードします。

解　説　GSAVE は、画面に表示されているグラフィックイメージを指定したファイルに保存するツールです。また、ファイルに保存された画像を画面に再表示するツールが GLOAD です。GLOAD のほうが DITHER よりも画像を画面に表示する速度が速いので、もう一度画面に表示したいイメージファイルは GSAVE で保存しておくとよいでしょう。ここで RENDER によるイメージファイルと GSAVE により保存されたファイルは異なるので注意してください。

実行例

```
gsave   sample.pic
gload   sample.pic
```

CLS3　グラフィック画面の消去

書　式　CLS3

機　能　CLS3 コマンドは、画面上に残ったグラフィック画面を消去します。

解　説　CLS3 コマンドは現在画面上に表示されているグラフィック画面を消去するツールです。テキスト画面も同時に消去します。

MODEL
モデルファイルから RENDER 用のデータを生成

書　式

MODEL　［−l］［−p　〈パラメータファイル〉］〈モデルファイル〉　……

機　能

MODEL コマンドは、モデルファイルから RENDER 用のデータを生成して標準出力に出力します。

オプション

−l　　画面上にワイヤフレームで表示。
−p　　パラメータファイルを指定。

解　説

MODEL コマンドによる結果は標準出力に出力されますので、リダイレクトを使って結果をファイルにしたり、パイプを使って出力結果を直接 RENDER に渡してイメージファイルを生成できます。

実行例

```
model  -l  -p  sample.001  sample.mod
model  -p  sample.001  sample.mod  |  render  sample.pic
```

MOTION
モーションファイルからパラメータファイルを作成

書　式

MOTION　〈モーションファイル〉

機　能

MOTION コマンドは、モーションファイルからモーションのパラメータファイルを作成します。

解　説

MOTION コマンドはあらかじめ作られたモーションファイルの中で指定されたパラメータに基づき、指定された数のパラメータファイルを作成します。このとき各ファイルの変数値は既存のパラメータファイル(モーションファイルの中で指定したもの)の変数値に依存して与えられます。MODE のモーションエディタでモーションデータを作成した場合は、パラメータファイルは自動生成されます。

実行例

```
motion  test.mot
```

Appendix F フレームバッファ

これからCGを本格的に始められる方、特に3次元の物体をモデリング、レンダリングし、リアルな画像をディスプレイ上で見てみたい方にとって「フレームバッファ」は必需品です。一度、フルカラーの画像を体験してしまうと、フルカラー以外の世界にはもどれないほどの魅力があります。その「フレームバッファ」について、NECのPC-9800シリーズを例に説明します。

PC-9800シリーズとフルカラー

■PC-9800シリーズの画面の色数

NEC PC-9800シリーズでは、標準での画面の色数は黒、白を含む原色の8色(デジタルRGB出力時)、またはRGBそれぞれ16段階で指定した任意の4096色中16色(アナログRGB出力時)です。ソフトの性格、用途によっては色数を多くするとかえって操作が複雑になり、使い勝手が悪くなる場合もあるので、標準での色数は多い方がよいと一概にはいえませんが、CGの出力には決して十分ではありません。

■フルカラー画像

一般にCGにおいてフルカラーとは、光の3原色RGBそれぞれを256段階表現できるものをいいます。256段階という値は、コンピュータのメモリの節約と周辺機材の性能、人間の眼の識別力を総合した妥協の数字です。世の中のすべての色や明るさを表現できるわけではありませんが、一般には十分な色数を表現できます。RGBは独立して値を持ちますので、色数は3色の組み合わせで256の3乗(256×256×256＝16777216)、約1677万色になります。

■フルカラーを出力する方法

まず、フルカラーの画像データを出力できるソフトであることを確認します。もちろん「MODE」はフルカラー対応です。「画像表示」や「画像連続表示」を実行した後にできる〝PIC″がフルカラーのイメージファイルです。

フルカラーの画像データをPC-9800シリーズのディスプレイに出力する方法は2つあります。1つは標準の色数を用いて擬似的に表す方法、もう1つはフレームバッファを加えて真にフルカラーの映像信号をディスプレイに送る方法です。

標準の色数を用いて近似する方法の1つにタイリング法があります。3×3、4×4ぐらいのピクセルの升目を想定し、そこに原色をいかに並べるかで見た目の色を中間色にする方法です。また、同様に画像の解像度を犠牲にしないでランダムな要素をより多く加えたディザリング法もあります。いずれの方法も、あくまでも擬似的に色数が多いように見せかける方法なので、細かい模様などがつぶれてしまうなどの欠点があります。ただし、絵柄によっては独特のざらついた質感がプラスの要素になるので、捨て難い点もあります。なによりも特別な機器が必要ないので、これらの方法は手軽にフルカラーを確認するにはよいでしょう。MODEでは「画像表示」や「画像連続表示」を実行すると、タイリングとディザリングを行って表示します。

フレームバッファを用いると、直接画像データをフルカラー画面で見ることができます。また、各種ペイントソフトを併用するとデリケートな色の変換や描き加えができます。ただし、PC-9800シリーズではオプションになりますので、ある程度の出費は覚悟しなくてはなりません。

フレームバッファの機能

■メモリ

フレームバッファ(Frame Buffer)はどのような働きをするのでしょうか。フレームバッファは別名フレームメモリ(Frame Memory)ともいわれ、画像メモリとしての機能を持ちます。各用語を簡単に定義すると、以下のようになります。

フレーム(Frame)	：	画面
バッファ(Buffer)	：	動作の異なる2つの装置の間にあって、速度、時間などの調整を行ったり、両者を独立に動作させたりするための記憶装置
メモリ(Memory)	：	記憶装置

フレームバッファとはコンピュータ本体と出力用ディスプレイの間に立つ、画面のための記憶装置のことです。

PC-9800シリーズは画面表示用のメモリとして、256Kバイトを持っています(一部機種を除く)。これはVRAMといわれ、次のように使われています。

・画面の解像度は横640×縦400ドット、全体で256000の画素を持つ
・それぞれの画素は16色、つまり4ビットの識別が可能
・画面は2画面

以上をコンピュータの記憶容量として計算すると、256000×4×2=2048000となり、2048000ビットの情報量になります。1ビットは、0か1かを区別する最小の単位です。一般には1バイト(8ビット)を1単位とするので、2048000÷8=256000バイト、256Kバイトと数えます(厳密には1024バイトで1Kバイト)。

それでは、フルカラーの画像を表示するためのメモリ、フレームバッファにはどれくらいの記憶容量がいるのでしょうか。以下に算出してみましょう。

・画面の解像度は横640×縦400ドット、全体で256000の画素を持つ(標準の画面出力と同じ)
・それぞれの画素はRGBごとに256色、つまり8ビットの識別をRGBごとに8×3=24ビット必要
・画面は1画面

全体の容量は768Kバイト(256000×24×1÷8=768000ビット)になります。

■D/Aコンバータ

メモリとしてのフレームバッファに存在している画像は単なる0と1の並び、いわゆるデジタル情報です。値として扱えても色や明るさを表す電気的な強弱にはなっていません。このデジタル情報を出力機器に対応するアナログ量に変換することをD/Aコンバートといいます。アナログ量とは連続的な変化を持つ、数値で直接指示できない量のことです。人間が感覚器官で捉えている刺激はほとんどアナログ量です。フレームバッファにはデジタル値としてのRGBの段階的な値をアナログ量に変換するD/Aコンバータの機能があります。

デジタル値は絶対不変ですが、アナログ量に同一のものはあり得ません。D/Aコンバータの性質によって、アナログ量も変化するはずです。フレームバッファのメーカー、あるいは調整によって色みや明るさが僅かに異なることもあります。

■映像信号出力

フレームバッファの第3の機能は、画像をディスプレイに表示する信号に変えることです。電気的なアナログ量も映像信号として出力しないと、ディスプレイに表示できません。

PC-9800シリーズ対応のディスプレイは、水平同期周波数が24.83KHzのアナログRGB信号を受像するように設計されています。水平同期周波数とは、映像信号が1秒間に画像を左から右に走査する回数のことです。一般のテレビなどでの映像信号(日本ではNTSC規格)では15.75KHzで、PC-9800シリーズとの互換性はありません。アナログRGB信号とはRGBの各信号を独立したアナログ量で伝える方式です。輝度から色情報までを1本の信号で伝えるコンポジットビデオ信号(一般のテレビなどの映像信号)との互換性はありません。

市販のPC-9800シリーズ用フレームバッファのほとんどは、PC-9800の標準アナログRGB信号に対応していますので、いま使用しているディスプレイにそのまま映像を映すことができます。また、フルカラーの映像の上に標準のアナログRGB信号やデジタルRGB信号をオーバーラップ(スーパーインポーズ)させて表示することもできます。

おもなフレームバッファ

NEC PC-9800シリーズ対応のフレームバッファの一部を紹介します。

ソフトによっては、一部のフルカラーペイントソフトのように特定のフレームバッファにのみ対応していますが、「MODE」はどのフレームバッファにも画像データを出力できます。

SuperFrame2/2Σ(サピエンス)

人気のフルカラーペイントソフト「SuperTableau」が動作するフレームバッファです。基本的表示機能が充実し、安定した性能で広いユーザーに使われています。

HyPER-FRAME/+(デジタルアーツ)

スタンダードなフルカラーペイントソフト「HYPER彩子」が動作するフレームバッファです。ハードウェア上のズーム機能やスクロール機能を駆使すると、簡単なアニメーション出力も可能です。

写像(キャスト)

手軽なフルカラーペイントソフト「写像ペイント」が動作するフレームバッファです。入手しやすい価格とハードウェア上での簡単なアニメーション出力機能などで人気を集めています。

MODEのフルカラー出力

前述したようにMODEでは「画像表示」や「画像連続表示」を実行すると、フルカラーのイメージファイル"PIC"を自動的に作成します。フルカラーで表示するときはフレームバッファに付属のツールを使います。以下に例を示しますので、詳しくはお使いになっているフレームバッファのツールのマニュアルを参照してください。

■SuperFrame2Σのツール

10章で作成した木のデータをフルカラー表示してみましょう。SuperFrame2Σのツールを使って表示するときは、次のように入力します。

```
B>both ↵                          … パソコン画面とスーパーフレームの同時表示を指定
B>fil2sf  j, 213, 133, 0, 0, pic ↵ …画像ファイルの内容をスーパーフレームに
                                     転送、PICはファイル名
```

■HyPER-FRAME+のツール

同じく10章で作成した木のデータをフルカラー表示してみましょう。HyPER-FRAME+のツールを使って表示するときは、次のように入力します。

```
B>hfon ↵                      … ボードの使用開始を宣言
B>hfsore  1 ↵                 … フレームバッファのソフトウェアリセットを設定
B>hfbnio  1 ↵                 … I/O方式のアクセス方式を選択
B>hfdisp  2 ↵                 … 画面表示をスーパーインポーズに指定
B>hfload  pic 0 0 213 133 0 ↵ … 画像ファイルのロード、PICはファイ
                                 ル名
```

Appendix G
作品例と部品

付属のディスクには本文中で説明したもののほかにも作品例(グラスと口絵作品「雪の結晶」)と部品(アルファベット)が収録されています。

■グラス

グラスは「保存用システムディスク」の¥GLASSディレクトリにあります。使用するときは¥GLASSディレクトリの内容をすべて別のディスクにコピーしてください。mode glass3 ⏎ を実行すると、グラスが読み込まれます。

■口絵作品「雪の結晶」

口絵作品「雪の結晶」が「保存用システムディスク」のYPICTUREディレクトリにあります。本作品はフルカラーのイメージファイルなので、フレームバッファがないとお使いになれません。ファイルは圧縮されていますので、次のような準備が必要です。

① MS-DOSを起動して「保存用システムディスク」をドライブA、初期化済みのディスクをドライブBに入れ(初期化の手順は「インストール」を参照)、次のように入力。

```
B>copy  a:¥picture  b: ⏎
B>yuki ⏎
```

※ YUKI.EXEはLHAの自己解凍形式ファイルで、実行するとYUKI.PICが作成されます。画面に表示するときはフレームバッファに付属のツールを使ってください。

■アルファベット

アルファベットのモデルファイルは「保存用システムディスク」の¥ALPHAディレクトリにありますので、必要なものをコピーして使ってください。アルファベットの一覧ファイルはALPHA−C.MOD(大文字一覧)とALPHA−S.MOD(小文字一覧)です。

索　引

英数字・記号

ア

カ

サ

■ 監修

太田　昌孝

1959年11月19日生まれ
東京大学　理学部情報科学科　博士過程修了
東京工業大学　総合情報処理センター助手

㈱生活構造研究所にてLispやCの処理系を作成後CGに転向、㈱SEDICの創立メンバーの1人となり「マンダラ1983'」やニュースセンター9時のオープニングを手がける。㈱フロッグスおよび太陽企画㈱にて、真にフレキシブルなCGシステムのありかたを探究。ネットワークへの造詣も深い。著書には『応用グラフィックス』(共著、弊社刊)がある。
趣味はまずなによりもコンピュータだがそのほか料理、ロールプレイングゲーム、スキーなどもたしなむ。さそり座、A型。

■ 原稿執筆

横山　弥生（「1章　数値と感性によるCGデザイン」、CG作成）

1960年3月8日生まれ
武蔵野美術大学　造形学部油絵学科卒業
トキワ松学園女子短期大学、専門学校桑沢デザイン研究所、文化学院芸術専門学校　講師

造形・デザイン教育へのコンピュータグラフィックスの応用研究に従事。Digital Image展などで作品発表。日本デザイン学会会員。日本図学会会員。
趣味は生け花、熱帯魚など。魚座、A型。

島田　信吾（「2章　アニメーション」、「Appendix F　フレームバッファ」）

1959年11月25日生まれ
多摩美術大学　美術学部建築科中退
青山CGスクール・メロン　教務担当

マルチメディアの実現を目指して在学中に開校準備中の青山コンピュータスクール・メロン(現：青山CGスクール・メロン)に参加。現在、4Dデザイン、映像機器ハードウェア、CGアニメーションの指導の傍ら番組タイトル、企業内VPなどのCGアニメーションの素材作りを行う。
趣味はマルチメディアクリエート、ハードウェアの構造解析。そのほかピアノ、自転車乗り、星をみること。射手座、A型。
NIFTY ID：GBF02465

■ MODE プログラム

丸川　一志
出口　哲生

■ ツールプログラム

石井　秀治	濱野　尚人
田口　景介	打越　浩幸
河村　博之	

■ CG 作成

矢口　英男	飯田　恒夫
渡辺　和年	鈴木　敏晃
五島　理江	東海　徳亮
中山　洋三	兼子　美弥子
森本　忠則	

■ 編集協力

佐々木　俊郎
藤沢　克樹
青柳　貴洋
八ッ代　裕美子

● 本書の内容に関するご質問は、小社第一書籍編集部まで、封書（返信用切手同封のこと）にてお願い致します。
電話によるお問い合わせには、応じられません。
なお、本書の範囲を越える質問に関しては、お答えできない場合もあります。
● 落丁・乱丁本は、送料当社負担にてお取り替え致します。
お手数ですが、小社営業部までご返送ください。

PC-9801 3次元CG作成システム MODE

1992年3月31日　初版発行
定価7,800円(本体7,573円)

監　修　太田 昌孝
編　集　アスキー書籍編集部
発行者　藤井 章生
編集人　佐藤 英一
発行所　**株式会社アスキー**
〒107-24　東京都港区南青山6-11-1スリーエフ南青山ビル
振　替　東京4-161144
大代表 (03)3486-7111
出版営業部 (03)3486-1977（ダイヤルイン）

制　作　株式会社 GARO
印　刷　大日本印刷　株式会社

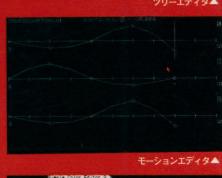
編　集　濱中 悟／辻 憲二

ISBN4-7561-0105-4　　Printed in Japan

● I3032

PC-9801
3次元CG作成システム
MODE

定価7,800円(本体7,573円)

ISBN4-7561-0105-4 C3055 P7800E

◆ 3次元CGアニメーションまで作成可能

◆ すぐ楽しめるサンプルファイルを収録

◆ 市販のフレームバッファに対応（1677万色同時表示可能）

◆ ソースリストが付属し、システムを拡張、改良可能

ポリゴンエディタ▲

■本パッケージの内容

"MODE"システムとは、『応用グラフィックス』(弊社刊)で紹介した汎用的なCGツール"MO｜RE"システムを発展させた3次元CG作成システムです。ポリゴンを作る"ポリゴンエディタ"機能、作ったポリゴンを配置する"ツリーエディタ"機能、動きを作る"モーションエディタ"機能などを使って編集し、カラーアニメーション表示ができます。

"MODE"システムは、特別なハードウェアを用意しなくてもお手持ちのパソコン(PC-9801)で手軽に本格的CGが楽しめます。ただし、お絵描きツールとは違い、CGの基礎的な知識が必要です。

付属のディスクパッケージには次のようなデータが収録されています。

ツリーエディタ▲

- 背景色(10色)と物体色(10色)ファイル
- アルファベットロゴのモデルファイル(大文字、小文字、計52文字)
- 作品例(5点)
- ソースコード付きツール(17件)

MODEシステムにはメインプログラムの"MODE"のほか、単体で使えるツールがあります。

- ・カラーエディタ
- ・平面エディタ
- ・円錐を作成
- ・球を作成
- ・正多面体を作成
- ・円柱を作成
- ・立方体の角をカット
- ・回転体を作成
- ・厚みを付加
- ・グラフィック画面の消去

モーションエディタ▲

■必要な機器構成

●パソコン：　NEC PC-9800シリーズ用。ただし、PC-9801/E/F/M/U/UV/VM/DO/DO+、PC-9801XA/LT/HAを除く(なお、PC-9801XL/XL2/RLおよびPC-H98ではノーマルモードでのみ利用可能)。

●メモリ：　メモリのフリーエリアが380Kバイト以上必要です。

●ディスプレイ：高解像度アナログRGBディスプレイでお使いになることをお勧めします(モノクロ表示では利用できない機能があります)。

●マウス：　バスマウス、シリアルマウスのどちらでも使用できます。

●ソフトウェア：日本語MS-DOS(Ver.2.0、3.1、3.3x)が必要になります。ご注意ください。

カラーエディタ▲

定価7,800円(本体7,573円)

ISBN4-7561-0105-4　C3055　P7800E